Cao DG, An B, Shi PC *et al.* Providing virtual cloud for special purposes on demand in JointCloud computing environment. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 32(2): 211–218 Mar. 2017. DOI 10.1007/s11390-017-1715-1

Providing Virtual Cloud for Special Purposes on Demand in JointCloud Computing Environment

Dong-Gang Cao [1], *Member, CCF, IEEE*, Bo An [1,*], Pei-Chang Shi [2], *Member, CCF* and Huai-Min Wang [2], *Member, CCF, IEEE*

[1] *Key Laboratory of High Confidence Software Technologies (Peking University), Ministry of Education Beijing 100871, China*

[2] *National Key Laboratory for Parallel and Distributed Processing, National University of Defense and Technology Changsha 410073, China*

E-mail: {caodg, anbo}@pku.edu.cn; shipeichang@kylinos.cn; whm_w@163.com

Received November 2, 2016; revised December 20, 2016.

Abstract Cloud computing has been widely adopted by enterprises because of its on-demand and elastic resource usage paradigm. Currently most cloud applications are running on one single cloud. However, more and more applications demand to run across several clouds to satisfy the requirements like best cost efficiency, avoidance of vender lock-in, and geolocation sensitive service. JointCloud computing is a new research initiated by Chinese institutes to address the computing issues concerned with multiple clouds. In JointCloud, users' diverse and dynamic requirements on cloud resources are satisfied by providing users virtual cloud (VC) for special purposes. A virtual cloud for special purposes is in essence a user's specific cloud working environment having the customized software stacks, configurations and computing resources readily available. This paper first introduces what is JointCloud computing and then describes the design rationales, motivation examples, mechanisms and enabling technologies of VC in JointCloud.

Keywords cloud computing, JointCloud, virtual cloud (VC), cloud working environment

1 Emerging of JointCloud Computing

1.1 From Single Cloud to JointCloud

Since the concept of cloud computing was proposed in 2006, cloud computing has been considered as the technology that probably drives the next-generation Internet revolution and rapidly becomes the hottest topic in the field of IT. The first generation (termed as Cloud 1.0 in this paper) is mainly focusing on aggregating large-scale IT resources into a single cloud provider and providing users with well-managed, auto-provisioned resources and services, while notably increasing IT resource utilization through service consolidation.

Cloud 1.0 has significantly reduced IT expense with always-on services. However, with the globalization of the economy, the cross-border trade of commodities and services is continuously expanding, leading to the increasing interdependence of world economies. The economic globalization calls for globalized Cloud services being provisioned in a geo-distributed manner at high availability and low cost. Several grand challenges raised with the globalization of Cloud services.

On one hand, a cloud vendor has to deploy datacenters across all over the world. This is similar to the way used in early airline companies to expand their services by adding flight courses to a destination country to provide globalized flight courses. However, many cloud-enabled world businesses usually demand a burst of computation that exceeds the remaining computing capacity of a single cloud. For example, many wholesale events like "November 11" from Taobao, "Black Friday" from Walmart usually require more than 10x

Regular Paper

Special Section on MOST Cloud and Big Data

This work is supported by the National Key Research and Development Program of China under Grant No. 2016YFB1000105 and the National Natural Science Foundation of China under Grant Nos. 61272154 and 61421091.

*Corresponding Author

resources compared with normal days, creating a huge pressure to a single cloud vendor, which either is not able to provide demanded resources, or has to provision IT resources according to the peak requirements, leading to notably increased cost and low IT resource utilization, which violates the essential goal of cloud computing to increase IT resource utilization.

On the other hand, under the newly emerged "shared economy", the globalized economy is also experiencing a new evolution that advocates cooperation among multiple potentially competing entities rather than monopolization. For example, Jack Ma, the founder of Chinese e-commercial giant Alibaba, has recently advocated to establish an electronic world trade platform (eWTP), with the goal to provide "an open platform for private enterprises and coordination among international organizations, governments and social groups which focus on the development of small- and medium-sized enterprises (SMEs) and trade as well". The Chinese government has also advocated "the Belt and Road Initiative" to focus on "connectivity and cooperation among countries primarily between the People's Republic of China and the rest of Eurasia". With more national entities involved in such globalized economy and computation, it is unlikely to have a "one ring to rule them all" approach such that a single giant cloud dominates the market, due to issues like government regulation, data privacy and political issues. Hence, such globalized yet cooperative business demands new cooperative computing models across the world. Actually, there has been several successful examples in the modern service area, among which a notable one is the airline alliance like SkyTeam, Star Aliance and OneWorld for airline services.

JointCloud is a recent key project funded by China's Ministry of Science and Technology as a part of the National Key Program for Cloud Computing and Big Data, which borrows the ideas from airline alliances and aims at empowering the cooperation among multiple cloud vendors to provide cross-cloud services via software definition.

Prior efforts like SuperCloud and InterCloud focus on the fusion of cloud services, usually via a third-party middleware. The middleware, as an overlay, invokes different clouds and provides a uniform interface to end-users. The clouds are actually unaware of the cooperation with other clouds. Different from existing multi-cloud models, JointCloud pays more attention to the direct collaboration among different clouds. It defines a series of rules and provides common services to enable collaboration among clouds. In JointCloud, clouds are independent while cooperating closely with one another. Just like global airline alliances Sky team and Star Alliance, there are many independent member airlines, and they work with one another closely.

As shown in Fig.1, there are two different parts in the JointCloud architecture: the JointCloud collaboration environment (JCCE) and the peer collaboration mechanism (PCM). JCCE contains several BlockChain-based services for enabling the cooperation among independent clouds. Based on JCCE, clouds can cooperate with one another, as long as these clouds implement a software-defined mechanism (named peer collaboration mechanism, PCM) and provide related APIs. A software-defined PCM includes three horizontal panels and one vertical information panel. The resource panel contains various software-defined resources. The control panel manages various software-defined resources and deals with the details of connection, storage and computation among clouds. The business panel handles the collaboration (e.g., transactions) between clouds. All the three panels provide standard open APIs, which provide information to the information panel. The information panel is designed to facilitate the communication in the three-panel architecture and communicate with JCCE (i.e., to report the collected information to JCCE, to obtain new information from JCCE).

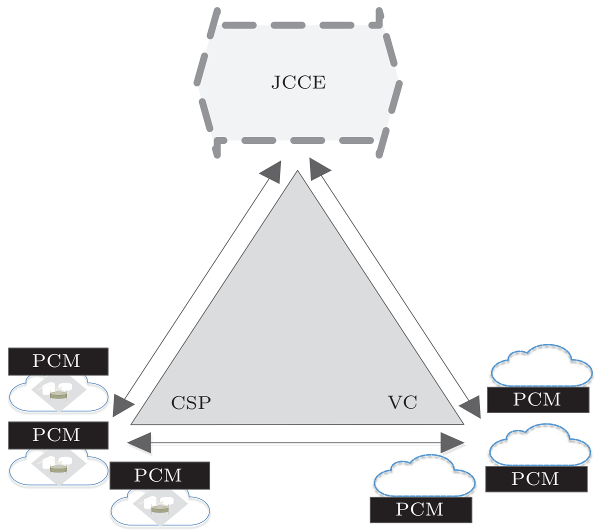

Fig.1. JointCloud architecture. CSP: cloud service provider.

There are also several virtual clouds (VCs) in Fig.1, which are customized clouds by aggregating resources and services of different clouds. Just like mobile virtual network operator[1] which provides mobile services through other operators' mobile networks, virtual clouds provide customized cloud services to end-users via other clouds' resources. Such virtual clouds can also implement the PCM to join the JointCloud ecosystem.

1.2 Virtual Cloud of Special Purposes for End-Users

With the development of cloud computing, more and more users prefer to run their applications on clouds such as big data processing, high-performance computing, and deep learning. However, users' requirements cannot be directly satisfied by current cloud computing service model. IaaS (Infrastructure as a Service) cloud only provides users with resources such as servers, storages, and networking. Users need to accomplish the environment installation with a complex configuration. In PaaS (Platform as a Service) cloud, users are provided with a cloud platform in which they can develop, manage, and deliver limited applications, but they usually cannot customize the runtime environment easily.

Virtual cloud for special purposes is a service of JointCloud, which aims to provide end users with a specific cloud working environment upon several clouds as shown in Fig.2, just like grid computing[2], which is the collection of computer resources from multiple locations to reach a common goal. A cloud working environment has users' readily available customized software stacks, configurations, and computing resources. Users can develop, test and run tasks in their working environment online through a web browser. Such a working environment is built upon a customized VC, which provides the most suitable resources from underlying clouds for the working environment. This environment could span multiple clouds seamlessly, and could help applications elastically scale out to temporarily use new resources from outside parties to deal with peak load problems.

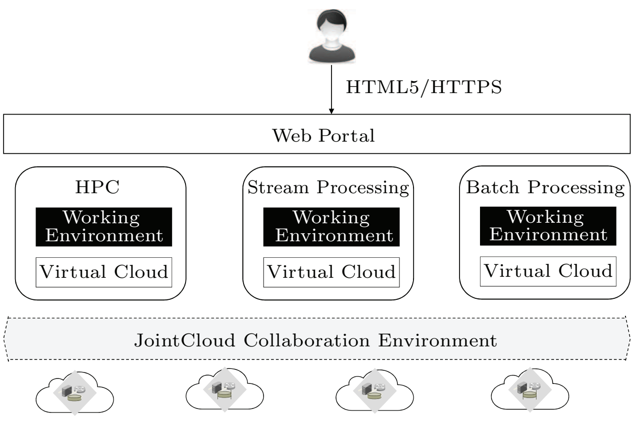

Fig.2. Virtual cloud for special purposes. HPC: hight performance computing

In the rest of this paper, we first present a motivation example for virtual cloud (Section 2). Then, we present our design of virtual cloud (Section 3), the current prototype system (Section 4) and a brief conclusion (Section 5).

2 Motivation Examples

2.1 Customized Working Environment

Many cloud applications are distributed and require a cluster of resources to run. Nowadays, clusters of commodity servers have become a major computing platform, powering both large Internet services and a growing number of data-intensive scientific applications. Researchers and practitioners have been developing an array of diverse cluster computing frameworks to simplify programming the cluster including MapReduce[2], Spark[4], Pregel (a specialized framework for graph computations)[5], and others[6-8]. However, installing and configuring a physical cluster for a specific computing framework is extremely difficult and highly depends on administrators' professional knowledge. Firstly, each component of the computing framework needs to be deployed on specific nodes of the cluster properly, which is quite complex and time-consuming. Secondly, the network should be configured correctly to connect every node into a complete system. Thirdly, distributed systems are faced with a typical problem of hardware failure, requiring administrators to set up a fault-tolerance mechanism. Fourthly, command line interface is usually the only way to operate the cluster, bringing considerable learning cost and inconvenience. Last but not least, purchasing or renting a physical cluster is too expensive for ordinary people to afford.

Virtual cloud for special purposes is designed for the aforementioned problem. To be specific, it provides users with cheap, flexible and easy-to-manage working environment, which is supported by virtual clusters in cloud environment, to run their own tasks. The deployment of working environment is much easier than physical clusters because of the virtual cloud's package mechanism. Virtual cloud can wrap users' working environment into a small package which can be deployed upon multiple clouds seamlessly. And the package can be shared with other users so that newcomers with little cluster deployment experience can directly select a proper package and deploy it to their own working environment in minutes or even seconds, then they can work on it just like on a well-configured physical cluster. On the other hand, virtual cloud has a powerful and easy-to-use web interface; thus users can conveniently manage their working environment, process tasks and view results through a web browser.

2.2 Cross-Cloud Scaling

Users' tasks vary significantly from one to another, thereby virtual cloud will automatically scale in or scale out according to the workload of the current task. However, sometimes the workload is so large that the whole cloud cannot afford. Under this circumstance, virtual cloud needs the ability to scale out to other cloud platforms to acquire more computation resources.

One of the problems is how to choose proper cloud resources for scaling. Cloud resources differ from one another in performances, service qualities, and billing rules, making users confused about which to choose. Another problem is how to scale working environment out to another cloud seamlessly. Manual migration of working environment is time-consuming and heavily relies on users' practical experience.

With the help of JCCE, virtual cloud provides an elegant solution to these two problems mentioned above. Virtual cloud is built on the JCCE and with the support of JCCE it can take over the management of various cloud infrastructures. Thus in the perspective of users, they are faced with a federated and unified cloud environment and can directly require or release cloud resources in the virtual cloud without visiting each cloud's management console. Besides, through the unified QoS metrics provided by JCCE, users can intuitively compare the homogeneous services from different vendors and choose the best one directly, no longer needing to test the various services one by one manually. As for the second problem, virtual cloud can sufficiently handle the migration with package mechanism and furthermore, all the instances out-scaled from the package of the original environment are guaranteed to be identical because the unified cloud environment has shielded the heterogeneity of different infrastructures.

3 Design of Virtual Cloud for Special Purposes

3.1 Design Goals

The goals of virtual cloud are to build a customized cross-cloud working environment conveniently for end users, increase the utilization of resources and guarantee the quality of service.

Customized Environment. Enterprises and individual users can pre-define all required resources according to their demands, including CPU, memory, network, operating system, software and required service. They do not need to know where infrastructures are located, and virtual cloud will provide a cloud working environment based on their requirements automatically. Then users could work in the cloud working environment through a web browser[9].

QoS Guarantee. Virtual cloud also takes the uncertainty into consideration which would significantly affect the efficiency of data provisioning, network transmission and computing. By using QoS metrics such as throughput, response time, data integrity and consistency, virtual cloud guarantees the stability and efficiency of working environment for users[10]. To be more specific, different cloud vendors supply their computing and storage resources at a different location, leading to a complicated circumstance of data transmission, network request latency, etc.[11-12] Therefore, it is essential for virtual cloud to make tradeoffs between transmission cost and computing efficiency, to ensure the customized QoS level[13-14].

Cross-Cloud Migration. Users may change their requirements and move their business to other clouds, thus in order to avoid vendor lock-in, virtual cloud allows to migrate working environment from one cloud to other clouds without any further installation and configuration, by leveraging software definition technology[15].

As a result, virtual cloud could help users easily build a unique, customized, cross-cloud and complicated working environment and quickly migrate it to other clouds whenever in need.

3.2 System Model

In order to satisfy the characteristics mentioned above, we present a virtual cloud system model conforming to the peer communication mechanism (PCM) as shown in Fig.3. The PCM is designed by JointCloud to facilitate the cooperation among clouds. Cloud vendors and virtual cloud can cooperate with one another, as long as they implement the PCM and provide related APIs. Our VC system model includes three horizontal panels: resource panel, control panel, business panel; and one vertical panel: information panel. The three-layer structure, just like different layers of the TCP/IP protocol, includes all agreements during cloud collaboration. The resource panel uses software definition tech-

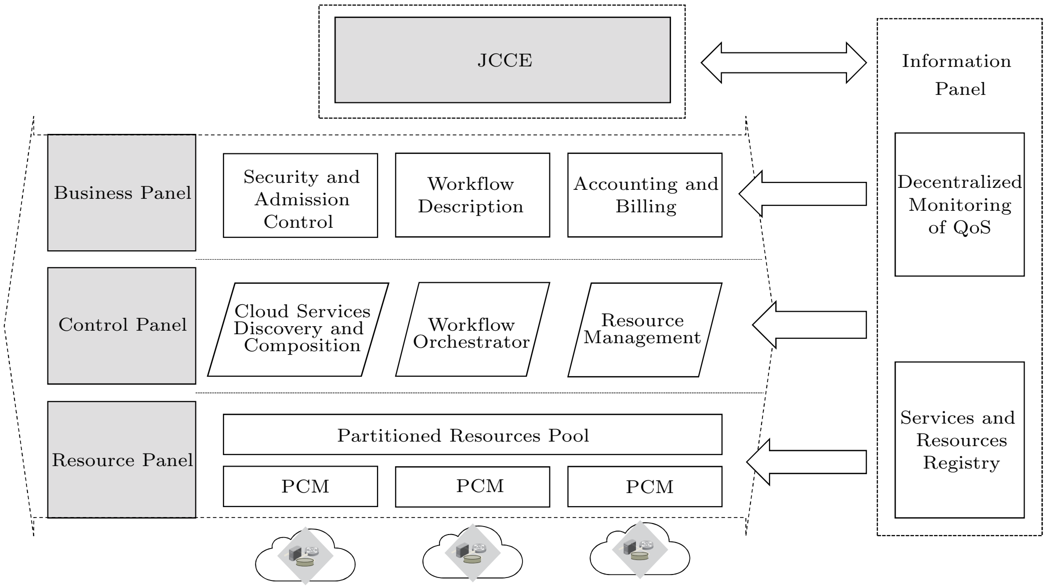

Fig.3. System model of virtual cloud.

nology provided by cloud vendors to make heterogeneous components abstract as a single resources pool. The control panel manages various software-defined resources and deals with the details of connection, storage and computing among clouds[16-17]. In specific, the control panel consists of several parts: cloud service discovery and composition, workflow orchestrator and resources management. Meanwhile, the control panel makes all resources transparent for developers on the business panel and provides a uniform interface for them.

The business panel handles the collaboration (e.g., transactions) between clouds. This panel is responsible for aggregating users' requirements, actions and feedbacks, cooperating with the resource panel and the control panel via software definition technology to automatically build, migrate and optimize the virtual environment. The business panel has three parts.

1) *Access Control Part.* Users could log in the system according to their username and password no matter where they are or which subsystem they want to visit[12,18]. Users do not need accounts of different cloud vendors but only one single account of virtual cloud[19].

2) *Workflow Description Part.* This part supports several interfaces for users to customize their unique environment requirements and special service demands like response time, given software. After customizing working environment, this part can calculate requirements of different resources, generate corresponding tasks and send the control panel orders to complete users' actions and update information.

3) *Accounting and Billing Part.* It interacts with the control panel and the resource panel timely through the information panel to get details of the working environment and keep consistency between infrastructure and user data[12,20-21]. Also, this part can record users' workload, resource consumption and the cost.

The information panel is designed to facilitate the communication among the three-panel architecture, register services and resources, record monitoring data of QoS, and communicate with JCCE. Also, the information panel stores information and messages published by all three panels. Therefore, the business panel, the control panel and the resource panel can communicate with each other and record events through this panel.

Virtual cloud could automatically build, manage, migrate and optimize a working environment based on users' requirement, by using software definition technology, making infrastructure transparent to users and scheduling resources to make a tradeoff between the quality and the cost.

Based on our system, users can build working environment quickly with their own specific requirements. Fig.4 shows an example on the procedure of the construction which consists of the following steps: 1) via browser and the workflow description part, users submit requirements including computing, networking, storage and software demands; 2) the business panel receives the description, generates deployment tasks and trans-

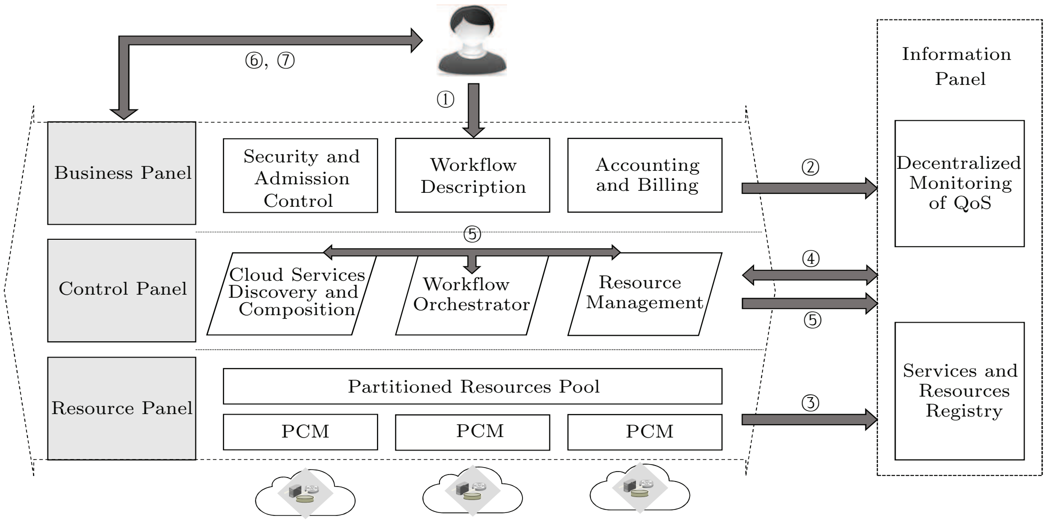

Fig.4. Procedure of construction.

fers these tasks to the control panel through the information panel; 3) the resource panel registers available resources to the information panel; 4) the control panel gets the tasks and then queries the information panel for available resources; 5) the control panel finds the best solution, builds environment, configures network, installs software needed, runs pre-defined scripts and reports the result to the business panel; 6) the business panel shows the result to users; 7) users work in the working environment using web browser, develop and deploy their application.

3.3 Software Definition Technology

As the system model is shown above, software definition is the key enabling technology to our virtual cloud system[22].

For our virtual cloud, we use several technologies to build the system. Firstly, we make use of container to package computing resources and achieve cross-cloud computation. The container is an operating-system-level virtualization method which allows multiple isolated user-space instances. The container imposes low overhead with high performance, so that it is useful for developers to set up a cross-cloud working environment. Nowadays, there are several kinds of containers (LXC, Docker, etc.) envolving quickly. We will choose one or more from them as our container to isolate and manage computing resources[23-25].

Secondly, we apply software-defined network (SDN) to manage and configure our network, and in specific, Open vSwitch (OVS) is chosen for its good support for OpenFlow, which is the enabler of SDN. SDN makes it possible for administrators to determine network services through the abstraction of lower-level facility, including dynamic, scalable computing and the path of network packets[26].

Thirdly, we provide a mechanism to wrap users' working environment into a package[27]. A package is aimed to describe the user-defined environment on virtual cloud, which can be conveniently shared by users. A user is able to submit a package to construct a specific environment automatically. The package is actually a compressed file that includes information data, such as scripts for creation, recovery and destroy, third-party binary files for special purpose and metadata for construction, management and monitor[28]. Also our mechanism provides users a convenient way to scale in or out their cloud working environment.

As described above, software definition technology is crucial to building our system. It works with low overhead, high efficiency, high operability and high flexibility, enables our system to manage, isolate, schedule and monitor computation, network and memory resources, supports for the building, migration and optimization functions, and makes virtual cloud as a service.

4 Status of Prototype System

We have designed and implemented an open source virtual cloud prototype system named Docklet①. It is

①https://github.com/unias/docklet/, Nov. 2016.

a starting point for our project. With Docklet, users can create a working environment to run their tasks through web browsers. The working environment can be stopped or restarted at any time, and can be shared to others. When tasks are running, users can see the status of the working environment in a webpage. Once the tasks are finished, the working environment can be destroyed and resources will be released.

A user view of Docklet is shown in Fig.5. Users can access resources via working environment, where they can perform tasks such as running applications, analyzing data, and visualizing analysis results in browsers. Each working environment is supported by a virtual cluster. A virtual cluster (Vcluster) is the unit for resource management and it consists of several virtual nodes (n) and a private network among them. Users can manage their jobs by creating, restarting, scaling in and out the virtual cluster with only a browser. Moreover, each working environment provides a portal for users to perform operations on virtual clusters. Nodes of the same working environment are in a private network, that is to say, nodes in one working environment cannot affect nodes in other working environment. A virtual node is, in fact, a Linux container (LXC)②.

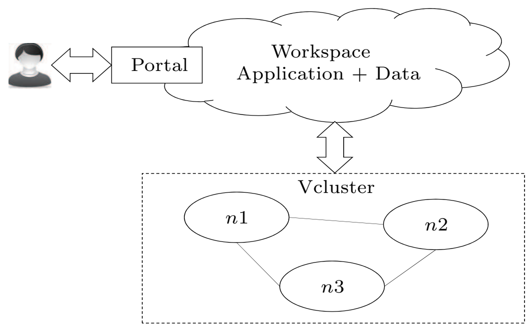

Fig.5. User view of Docklet.

Currently, Docklet has put into operation as a data processing platform③ in Peking University (PKU), providing cluster computing resources to teachers and students for education and research. Users can log in Docklet with their PKU account and create a working environment. Then they can write program interactively in the browser and see the running results in real time. They also can upload their pre-written program and raw data to run the job. Moreover, Docklet has pre-prepared packages with data processing tools from R and python community, and frequently-used frameworks such as Hadoop and Spark, so that users can acquire a well-configured processing environment instantly and work for their job directly.

5 Conclusions

In this paper, we presented virtual cloud for special purposes, which aims to provide end users with a specific working environment upon several clouds. Our current prototype system Docklet has covered part of demands of virtual cloud. Users can create a working environment to perform specific tasks and can achieve the rapid deployment and migration of working environment by the package mechanism.

The research of virtual cloud has just started, and there is a five-year roadmap. Currently Docklet can only work upon a physical cluster in one cloud, which means all the resources of a working environment have to be located at a same physical cluster, making it difficult to support cross-cloud computation. In future research, we intend to investigate efficient mechanisms to implement the proposed system and several key issues including efficient aggregation of cross-cloud resources, fast construction of users' working environment, and a decentralized monitor of users' tasks, etc.

References

[1] Kiiski A, Hämmäinen H. Mobile virtual network operator strategies: Case Finland. In *Proc. the 15th ITS Biennial Conference*, Sept. 2004.

[2] Foster I, Zhao Y, Raicu I, Lu S. Cloud computing and grid computing 360-degree compared. In *Proc. Grid Computing Environments Workshop*, Nov. 2008.

[3] Dean J, Ghemawat S. MapReduce: Simplified data processing on large clusters. In *Proc. the 6th Symposium on Operating Systems Design & Implementation*, Dec. 2004, pp.137-150.

[4] Zaharia M, Chowdhury M, Franklin M J, Shenker S, Stoica I. Spark: Cluster computing with working sets. Technical Report No. UCB/EECS-2010-53, University of California, May 2010.

[5] Malewicz G, Austern M, Bik A *et al.* Pregel: A system for large-scale graph processing. In *Proc. the 28th PODC*, Aug. 2009, pp.135-146.

[6] Logothetis D, Olston C, Reed B, Webb K C, Yocum K. Stateful bulk processing for incremental analytics. In *Proc. the 1st ACM Symposium on Cloud Computing*, June 2010, 51-62.

[7] Ekanayake J, Li H, Zhang B, Gunarathne T, Bae S H, Qiu J *et al.* Twister: A runtime for iterative MapReduce. In *Proc. the 19th ACM International Symposium on High Performance Distributed Computing*, June 2010, pp.810-818.

②http://lxc.sourceforge.net/lxc.html/, Nov. 2016.

③http://iwork.pku.edu.cn, Nov. 2016.

[8] Murray D G, Schwarzkopf M, Smowton C, Smith S, Madhavapeddy A, Hand S. CIEL: A universal execution engine for distributed dataflow computing. In *Proc. the 8th ACM/USENIX Symposium on Networked Systems Design and Implementation*, Mar.30-Apr.1, 2011, pp.113-126.

[9] Wei Y, Blake M B. Service-oriented computing and cloud computing: Challenges and opportunities. *IEEE Internet Computing*, 2010, 14(6): 72-75.

[10] Yau S S, Huang D. Development of situation-aware applications in services and cloud computing environments. *International Journal of Software and Informatics*, 2013, 7(1): 21-39.

[11] Wu H, Zhang W, Zhang J, Wei J, Huang T. A benefit-aware on-demand provisioning approach for multi-tier applications in cloud computing. *Frontiers of Computer Science*, 2013, 7(4): 459-474.

[12] Wang Y J, Sun W D, Zhou S, Pei X Q, Li X Y. Key technologies of distributed storage for cloud computing. *Journal of Software*, 2012, 23(4): 962-986. (in Chinese)

[13] Zhang S, Qian Z Z, Wu J, Lu S L. Service-oriented resource allocation in clouds: Pursuing flexibility and efficiency. *Journal of Computer Science and Technology*, 2015, 30(2): 421-436.

[14] Wang P, Huang Y, Li K *et al*. Load balancing degree first algorithm on phase space for cloud computing cluster. *Journal of Computer Research and Development*, 2014, 51(5): 1095-1107. (in Chinese)

[15] Yu B, Han Y, Yuan H, Zhou X, Xu Z. A cost-effective scheme supporting adaptive service migration in cloud data center. *Frontiers of Computer Science*, 2015, 9(6): 875-886.

[16] Wang Z J, Zheng Q S, Cao J. Efficient coordinator in hybrid cloud. *Computer Science*, 2015, 42(1): 92-95, 105. (in Chinese)

[17] He J, Bresler M A, Chiang M, Rexford J. Towards robust multi-layer traffic engineering: Optimization of congestion control and routing. *IEEE Journal on Selected Areas in Communications*, 2007, 25(5): 868-880.

[18] Lin L, Hu J, Zhang J. Packet: A privacy-aware access control policy composition method for services composition in cloud environments. *Frontiers of Computer Science*, 2016, 10(6): 1142-1157.

[19] Niu S, Tu S, Huang Y. An effective and secure access control system scheme in the cloud. *Chinese Journal of Electronics*, 2015, 24(3): 524-528.

[20] Wang H. Privacy-preserving data sharing in cloud computing. *Journal of Computer Science and Technology*, 2010, 25(3): 401-414.

[21] Li X J, Wu Y, Liu X, Cheng H M, Zhu E Z, Yang Y. A novel datacenter-oriented data placement strategy of scientific workflow in hybrid cloud. *Journal of Software*, 2016, 27(7): 1861-1875. (in Chinese)

[22] Nie H, Yang X J, Liu T Y. Software-defined cluster. *Journal of Computer Science and Technology*, 2015, 30(2): 252-258.

[23] Seo K T, Hwang H S, Moon I Y, Kwon O Y, Kim B J. Performance comparison analysis of Linux container and virtual machine for building cloud. *Advanced Science and Technology Letters*, 2014, 66: 105-111.

[24] Merkel D. Docker: Lightweight Linux containers for consistent development and deployment. *Linux Journal*, 2014, 2014(239): 2.

[25] Bernstein D. Containers and cloud: From LXC to Docker to Kubernetes. *IEEE Cloud Computing*, 2014, 1(3): 81-84.

[26] Sridharan M, Greenberg A, Venkataramiah N, Wang Y, Duda K, Ganga I, Tumuluri C. NVGRE: Network virtualization using generic routing encapsulation. https://tools.ietf.org/html/rfc7637, Feb. 2017.

[27] Cui W, Zhan H, Li B, Wang H, Cao D. Cluster as a service: A container based cluster sharing approach with multi-user support. In *Proc. IEEE Symposium on Service-Oriented System Engineering* (*SOSE*), Mar.29-Apr.2, 2016, pp.111-118.

[28] Sun D W, Chang G R, Gao S, Jin L Z, Wang X W. Modeling a dynamic data replication strategy to increase system availability in cloud computing environments. *Journal of Computer Science and Technology*, 2012, 27(2): 256-272.

Dong-Gang Cao is an associate professor at Software Institute, School of Electronics Engineering and Computer Science, Peking University, Beijing. He received his Ph.D. degree in computer software and theory from School of Electronics Engineering and Computer Science, Peking University, Beijing, in 2004. His research interests include system software, parallel and distributed computing.

Bo An is a Ph.D. student in the Key Laboratory of Hight Confidence Software Technologies and School of Electronics Engineering and Computer Science, Peking University, Beijing. His research interests include cloud computing and distributed system.

Pei-Chang Shi is an assistant professor at National University of Defense Technology (NUDT), Changsha. He received his Ph.D. degree in computer software and theory from NUDT, Changsha, in 2012. His research interests are operating systems and distributed computing technology.

Huai-Min Wang is a professor at National University of Defense Technology (NUDT), Changsha. He received his Ph.D. degree in computer science and technology from NUDT, Changsha, in 1992. He has been awarded the "Chang Jiang Scholars Program" professor by Ministry of Education of China, and the Distinct Young Scholar by the National Natural Science Foundation of China (NSFC), etc. His research interests are middleware, software agent, trustworthy computing, and distributed computing technology.