

Research on an Approach of Cloud Workloads Deployment to Public Cloud Based on Open Source Standard

Xu Weixiang, Jia Lin

*School of Traffic and Transportation, Beijing Jiaotong University
No.3 Shangyuan Residential, Haidian District, Beijing City, China
wxxu@bjtu.edu.cn; 12121020@bjtu.edu.cn*

Abstract

Cloud computing is a kind of formula mode that is based on internet, and in this way, the sharing software and hardware resource and information can be provided for the computer and the other terminal devices on demand. However, due to technical differences among main service vendors, there is a compatibility issue on appropriate format that is based on the virtual machine manager of different cloud platforms. Therefore, it brings some hinder to flexible deployment of cloud workloads to some extent. While with the great development of open source standers, such issues can be solved in another train of thought. In this paper, it proposes a constructive solution to deploy cloud workloads based on open source standers. In the whole structure, the key point is that a cloud middleware is introduced and the application of MapReduce algorithm. The cloud middleware contains four kinds of modules, namely, OVF converter, classifier, interface matched and security check. And the improved MapReduce algorithm is used to the classification phase of the cloud workloads. In the program, cloud workloads are sent to the cloud middleware, and in the process of "classified → packaged → matching interface → security check → deployment", which can make cloud workloads deployed easily and flexibly. And this paper will elaborate the progress of each section through a theory case. The biggest construction of this program is manifested in four aspects that are function modular, operation independence, security and scalability. This program must be further improvement, based on the addition of functional roles and the development of open source standards, and has a strong theoretical and practical significance.

Keywords: *cloud workloads; deployment; cloud middleware; OVF; MapReduce*

1. Introduction

With the flourishing development of cloud computing and virtualization technologies, an increasing number of cloud workloads need to deploy on the different cloud datacenters. Nevertheless, those technological disparities between the existing major cloud vendors gives rise to the low efficiency of the cloud workload deployment on heterogeneous platforms, which makes the users' need cannot be met. Currently, many scholars have studied this issue from different aspects at some degree. Literatures [2], [3] and [6] make a deep research to avoid vendor locked-in from the perspective of OVF and open interfaces (OCCI, CDMI), and gives a comparative analysis on OVF, OCCI and CDMI. Literatures [4] and [5] have a deep research on the application of virtual machines based on open-source platforms and cloud datacenters. Literature [7] makes a study on deploying component, applications, and load balancing, which is based on open-source standards, and the advantage of its program is that the application deployment is independent of cloud service providers. Literature [8] describes that a private cloud platform designs in-house one monitoring system, which gives the idea of the application

of open source standards. Literature [9] has a deep research on the public cloud deployment, and the core features of its approach are easing security configuration, customer-driven and by using OVF to avoid being locked-in. Literature [10] makes an outstanding contribution of the VM's creation and management. Literature [11] puts forward an effectively approach to solve the resource allocation problem, and which is modeled it as a distributed constraint optimization problem. Literature [12] makes a deep research on workload management for virtualized resources, and it proposes a new model that is used to improve the system throughput, efficiency and utilization. Literature [13] has a deep research on the resource allocation problem, which puts forward a kind of mechanism of the admission control and scheduling, aimed at maximizing the source utilization and the quality of service. Those jobs make a deep study on the application of open standards and the management of virtual machines, however, it may need a further study on the specific program and process of cloud workloads based on the open source standards. Against the background of the deployment issue of cloud workloads, this paper analyzes the virtualization market and the mobility status quo of cloud workloads; then open standards for cloud computing are introduced, including open virtualization format standard (Open Virtualization Format, OVF), and open source interface standard (OCCI, CDMI, DeltacloudAPI); above all, on the basis of open source standards and related technologies, this paper projects a constructive solution on cloud workloads deployment; Finally, a theoretical case study is used by the further research. This approach emphasizes the role of the cloud middleware on the basis of open source standards and related technologies, the improved MapReduce algorithm is used to the classification phase of the cloud middleware, and make a deep research on the reaching OVF packages, matching interfaces and security policies of the cloud middleware, which has the workloads packaged separately and independently of the public cloud providers, and avoids being locked by a single vendor.

2. Analysis on the Issue of the Cloud Workload Deployment between Different Virtual Machine Managers

Currently, many of virtual disk formats are proprietary, such as Amazon AMI, VMware VMDK, Microsoft VHD, etc., which makes interoperability between different cloud datacenters temporarily hinder play. Amazon AMI (Amazon Machine Image) is the proprietary format of Amazon EC2 virtual machine, which is based on Xen virtual machine builder, and does not support the standard virtual machine on it to run. The user wants to use Amazon EC2 cloud service, you must create your own AMI, then package applications, operating systems, virtual machines, and other parts, but if you want the workload that is packaged by the Amazon files to deploy elsewhere, it requires a corresponding conversion technologies; otherwise this workload is "immovable". Similarly, as to the X86 virtualization technologies, Microsoft and Citrix's VHD format is not compatible with the virtual machine manager of VMware's VMDK ESX. The user wants to deploy workloads elsewhere, there must be corresponding conversion technologies, too.

3. Related Technologies and Open Source Standards

In this paper, the constructive program is based on the open source standards and related technologies, including virtual devices, open virtualization format (OVF), open source interface standards. Among them, the virtual device is one of the key technologies of the cloud workload transfer.

3.1. Open Virtualization Forma

OVF is a neutral format that is designed by Distributed Management Task Force (DMTF) [1]. Virtual machines can only use this format to migrate, but not to run. Virtual devices that are packaged by OVF have the advantage of security, portability, and scalability. OVF packages and OVF envelops are the main elements of the OVF overall.

The OVF package has the role of different files, as shown in Figure 1, including the OVF file (.ovf), one or more virtual machine image file, other resource files, a certificate file (.cert) and a summary list of files (.mf). While there must be the .ovf and image file, and other files are optional.

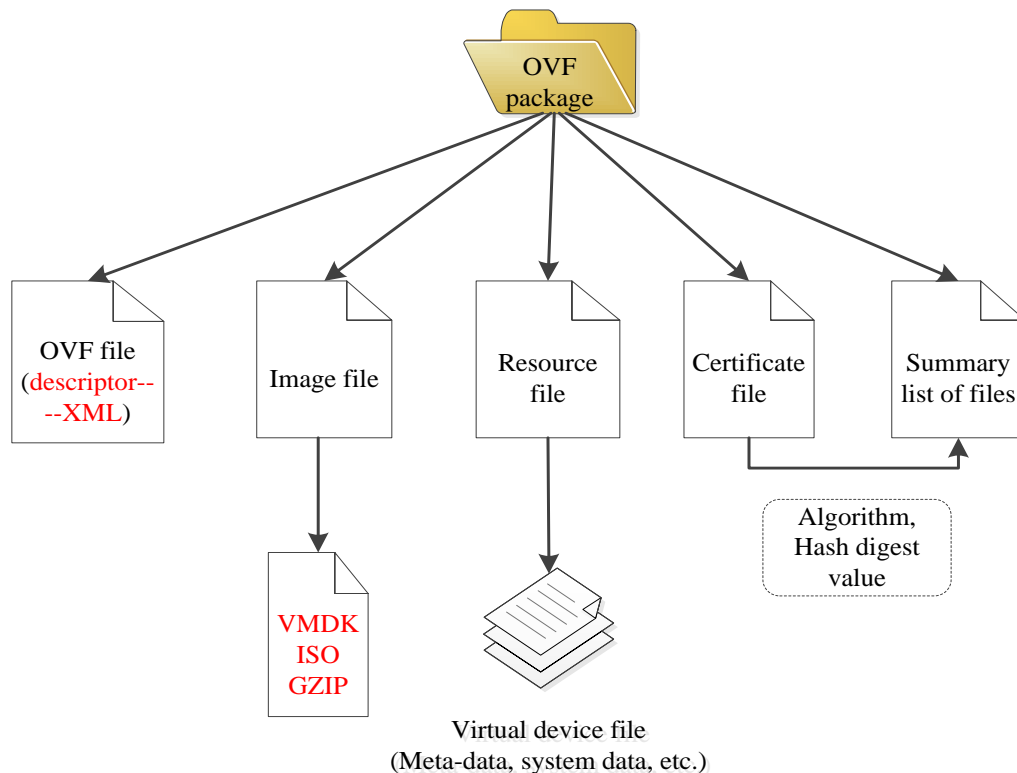


Figure 1. The Descriptor of OVF Package

In the OVF package, the .ovf file is a descriptor that is used to describe an integral virtualization solution and the characteristic of each part; the .image file is the binary file of the virtual device number (such as VMDK, ISO, GZIP, etc.), supporting multiple virtual machine image formats; the resource files are the files that associated with the virtual device, including metadata and system data; the .mf documents record hash digest values and algorithms of the OVF package files; the .cert file is a signature summary of a .mf file.

One OVF envelope is one core file of the virtual device. Users define related data of virtual devices in the OVF envelope, mainly including disk modules, start modules, network modules, and virtual machine modules, as shown in Figure 2. The virtual machine module contains a number of modules belonging to the virtual machine, such as operating system modules, license agreement modules.

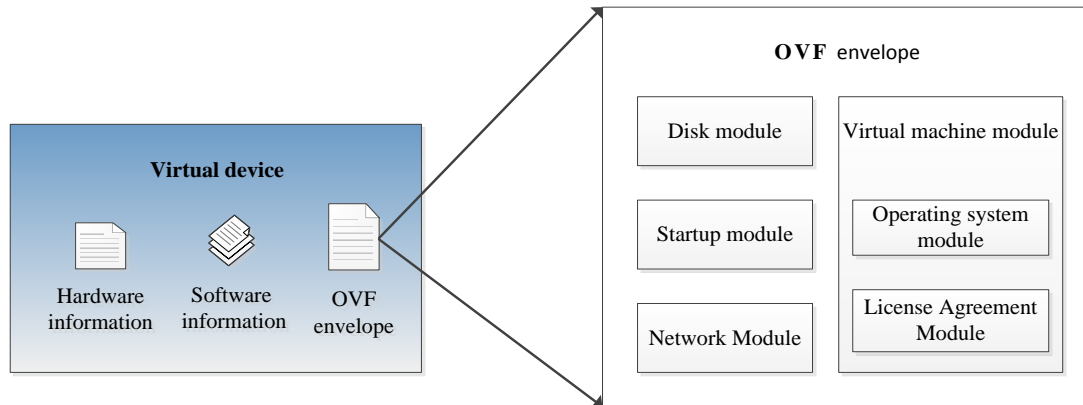


Figure 2. OVF Envelope

3.2. Open Source Interface Standards

Open source interface standards provide an interoperable means for different cloud platforms, including deploying applications, elastic expansion, storage, etc., which greatly ease the situation of users locked-in by vendors. Currently, there are three main types of cloud computing open interface standards, namely: OCCI (Open Cloud Computing Interface), CDMI (Cloud Data Management Interface), DeltacloudAPI.

OCCI that is studied by the OGF (Open Grid Forum) is used to provide cloud computing standardized API, and is a service-oriented architecture, which takes HTTP as the communication protocol.

CDMI is an interface that is used to deploy cloud storage systems. Comprehensive metadata model is the key element of the CDMI technology, providing a complex data structure.

DeltacloudAPI is developed by RedHat, and is used to realize the migration between the private and public cloud.

4. The Deployment Program Based on Open Source Standards

Based on the open source standards, the constructive program is proposed. As it can be seen in the Figure 3, there are three parts in the program, namely, three types of users, a cloud middleware and the public cloud. Users send the cloud workloads to the cloud middleware, and the work of deployment is completed by the cloud middleware.

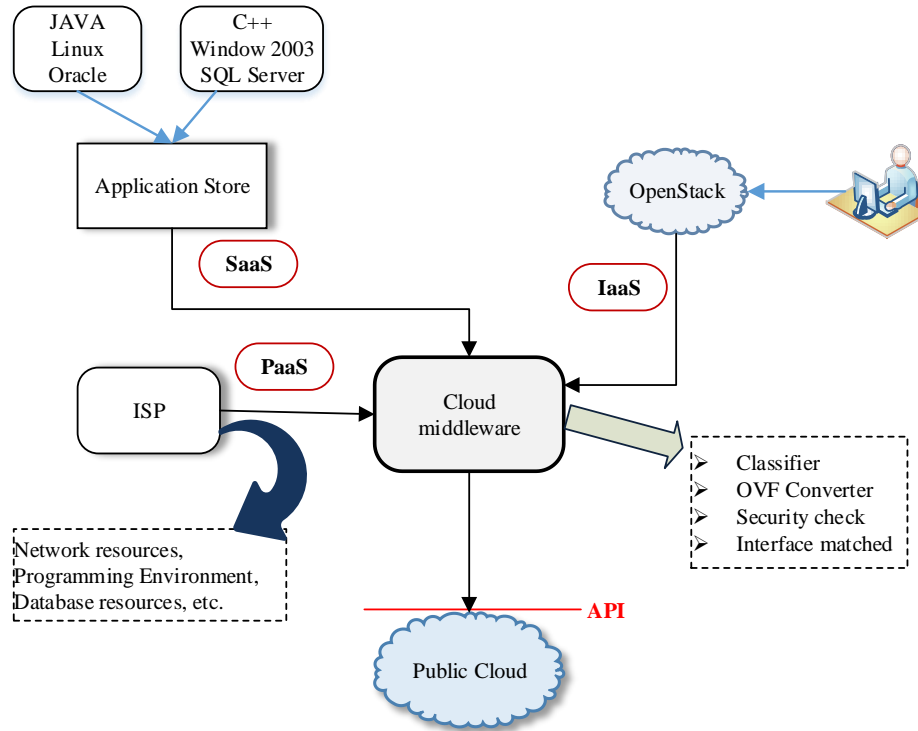


Figure 3. The constructive Program on the Deployment of Cloud Workloads

4.1. Users

The three types of users are ①SaaS vendors, ②PaaS vendors, ③IaaS private cloud enterprises, as shown in Figure 3.

(1) IaaS private cloud enterprises.

Enterprises build their own private cloud according to their own features, and employees access the private cloud through several terminals, such as desktops, tablet PCs, smart phones, and so on. Then the enterprises deploy their workloads to the public cloud based on needs.

(2) PaaS vendors.

As can be seen in the Figure 3, ISP is one example of PaaS vendors, and provides the network resources, programming environment and so on. PaaS vendors may need to deploy additional workloads on the public cloud.

(3) SaaS vendors.

A representative application store is one of SaaS vendors. There is no direct relationship among IaaS, PaaS and SaaS, while parts of the service of SaaS may run on the IaaS platform, so the SaaS vendor is one type of the users.

4.2. Public Cloud

The target cloud determines whether to run those virtual devices based on the examination result of the security. When receiving the stopping service notification of users, the public cloud sends virtual devices to users directly or to the cloud middleware.

Generally, it is good that one public cloud can provides a kind of virtualized environment, which can make users build the sharable and repeatable VMs. As is shown in the Figure 4, users can build the VMs through the VM builder, and the generated VMs can be saved for two formats, namely packaging and attachment. What is more, the

virtualized environment contains of user interface, web service layer and substrate, and the substrate will save the VM that is from users.

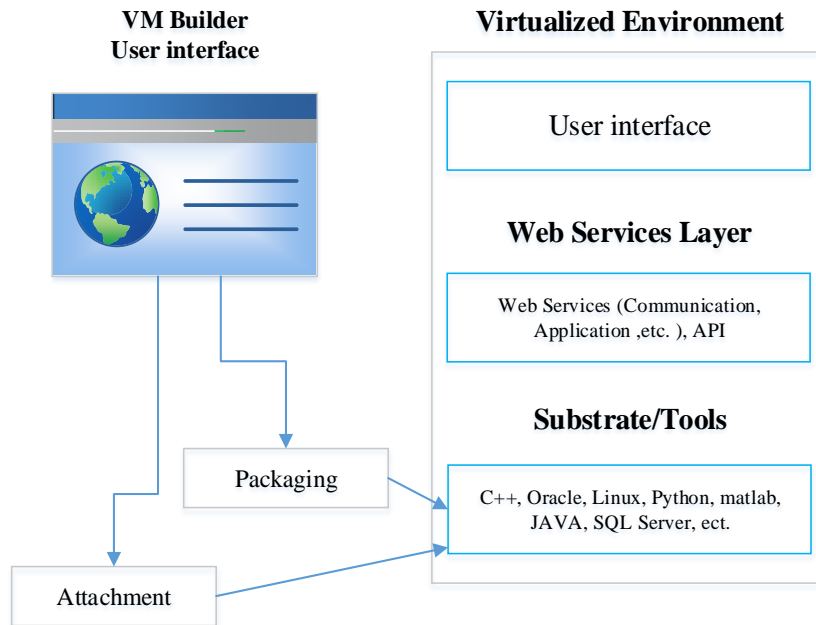


Figure 4. Virtualized Environment

4.3. Cloud Middleware

The cloud middleware has four function modules, including classifier, OVF converter, security check and interface matched.

It will examine whether there is the available VM in the public cloud at first. The cloud middleware will send the sharable and repeatable attachment to users. Otherwise, it receives the workloads that is from different users and complete the operation of OVF packaged and matching interfaces. In this paper, we will focus on the second case.

5. Theoretical Case Study

When users send workloads to the cloud middleware, the cloud middleware will generate five kinds of files, including OVF envelope, OVF package, a descriptor file of the target format, an information file of the security check, and a parts list file. As shown in Figure 5, then the cloud middleware sends those files to the target cloud.

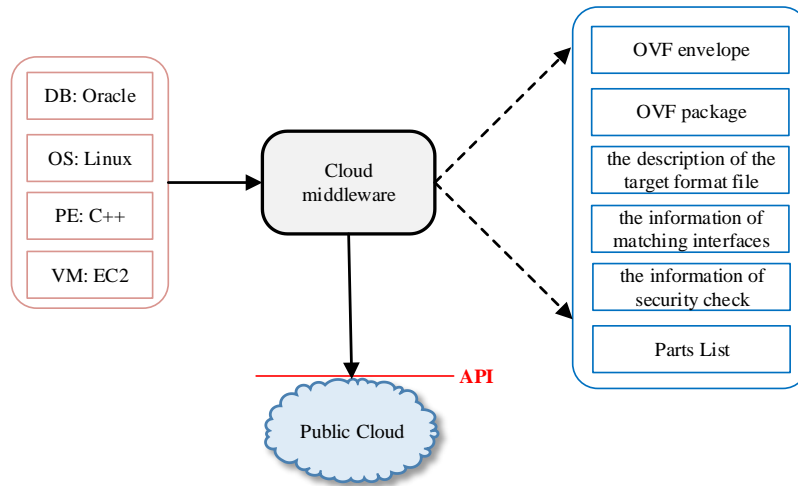


Figure 5. The Descriptor of Progress

The cloud middleware has two main functions that it receives cloud workloads and converts them into the OVF and that it matches the interfaces between open source interface standards and the target interfaces. As is shown in Figure 6, security and message modules control the process to operate or stop. The message module is responsible for connecting users and the target cloud, and passing information among users, the cloud middleware, and the target cloud. The security module provides a kind of appropriate security mechanism according to the information that is from the message module.

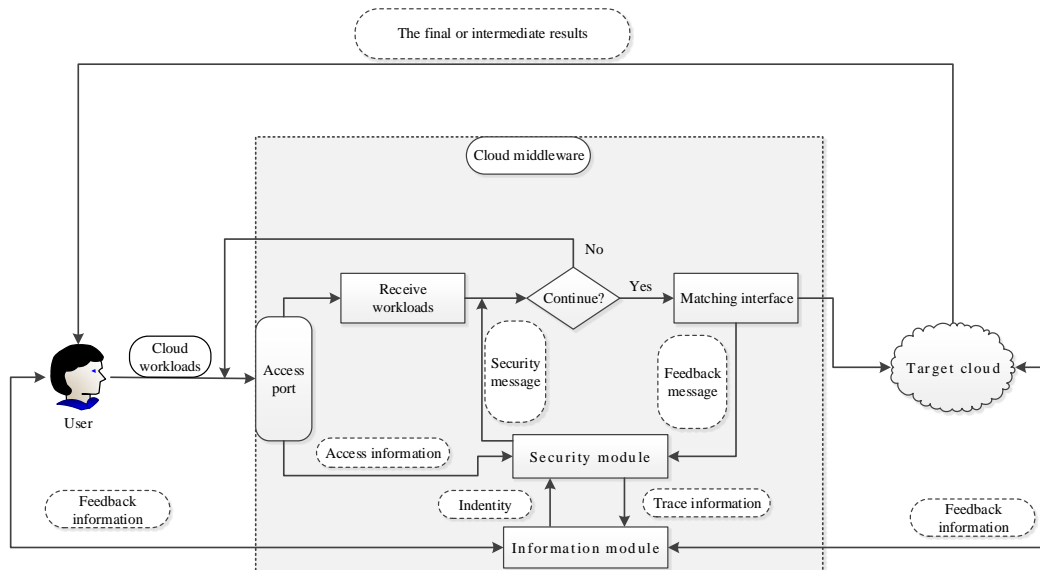


Figure 6. Virtualization Solutions

As can be seen from Figure 6, the cloud workloads are sent to the procedure through the access port. Firstly, the access port sends the security information to the security module, and the cloud middleware reads the files of the workloads and convert the workloads into the OVF. Secondly, the security module receives the identity information of the target cloud, which comes from the message module, and sends the related security information (security levels, identity information, etc.) to the procedure after converting the format. The security module determines that whether the matching procedure is run

based on the authentication information that is the users' and the target clouds'. Moreover, the qualified virtual devices are sent to the matching module, and the feedback information is sent back to the security module by the matching interface module and sent to the users and the target cloud by the message module. Finally, the cloud middleware sends the qualified workloads to the target cloud. Users send service requests to the target cloud through the message module of the cloud middleware, and the target cloud may send the intermediate or final results to users directly.

5.1. Cloud Workloads Classification

Virtual devices need to be packaged into OVF before them shifted to the public cloud. That work can be completed by enterprises, individual consumers, but also the cloud middleware. The cloud middleware receives the users' virtual devices, and separately packages the virtual devices that are from the same user, but different requirements.

In this paper, the MapReduce data analysis has been used, as to improve deployment efficiency, which is used to pack and classify OVF package. The data analysis that is based on MapReduce parallel framework can be directly analyzed and utilizes the model of mobile computing rather than moving data, so that it can make analysis of delay minimum. The specific process is shown in the Figure 7.

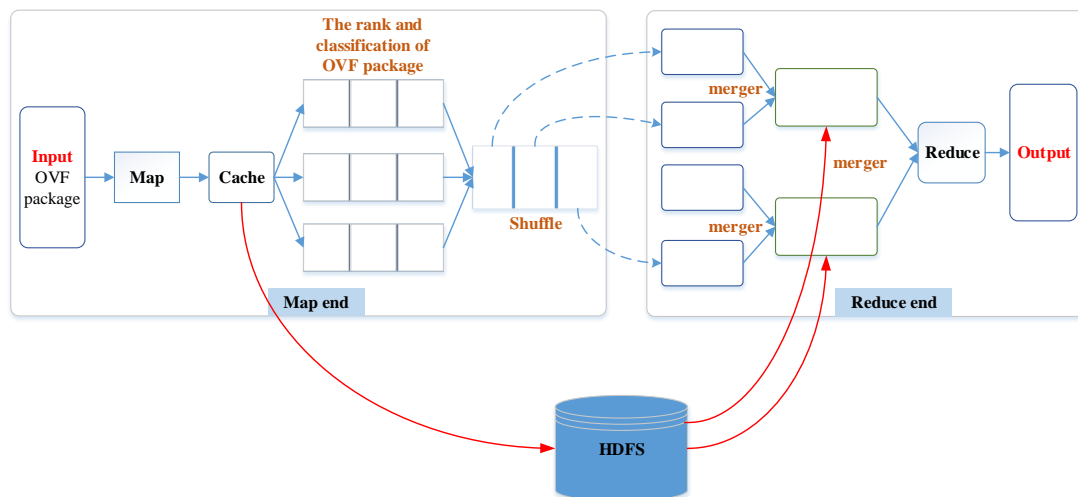


Figure 7. OVF Classification

In the Map phase, OVF packages are input, and the map processing program identifies the .ovf file, which describes virtual devices and their relationships in the OVF package. While the new OVF package that is none in the HDFS, then the different demand and classification of OVF packages will be sent to the Shuffle. It is worth noting that the number of OVF package that is sent to the Reduce end is one.

In the Reduce phase, firstly, the shuffle result set will be sent to the Reduce end, and the same classification can be merged to the same result set. Accordingly, HDFS, in the cluster, sends the rest demand of OVF packages to the intermediate result in the Reduce phase. In the last, the packed OVF and destination address files will be output.

What is more, the Cache processing program takes use of the needs updating algorithm, so that it can carry out communication between the Cache and HDFS. The pseudo-code is shown as follows.

Algorithm Cache-MapReduceOverview

```

Require: changes:  $\Delta \leftarrow (-\theta_1, +\theta_2)$ 
1: /*Process new input  $+\theta_2$  by running Map tasks*/
2: for  $i = T_{end}$  to  $T_{end} + (+\theta_2)$  do
3:  $V_i(\{k\}) \leftarrow \text{run\_maptask}(i)$ ;
4: end for
5: /*Propagate  $\Delta$  using Cache*/
6: for all key k do
7: /*Delete Map outputs for
*/
8: for  $i = T_{start}$  to  $T_{start} - (-\theta_1)$  do
9: Cache.Delete( $V_i(\{k\})$ );
10: end for
11: /*Insert Map outputs for  $+\theta_2$ */
12: for  $i = T_{end}$  to  $T_{end} + (+\theta_2)$  do
13: Cache.Insert( $V_i(\{k\})$ );
14: end for
15: /*Perform change propagation*/
16: Cache.Update(k);
17: end for
18: /*Adjust HDFS for the next incremental run*/
19:  $T_{start} \leftarrow T_{start} - (-\theta_1)$ ;
20:  $T_{end} \leftarrow T_{end} + (+\theta_2)$ ;
    
```

5.2. Matching Interfaces

Each cloud service provider has a specific interface plug-in. The cloud middleware achieves the work that matches open interfaces and target interfaces through the plug-in mechanism, packages its technical details and makes it independent of the target cloud. The specific process is shown in Figure 8.

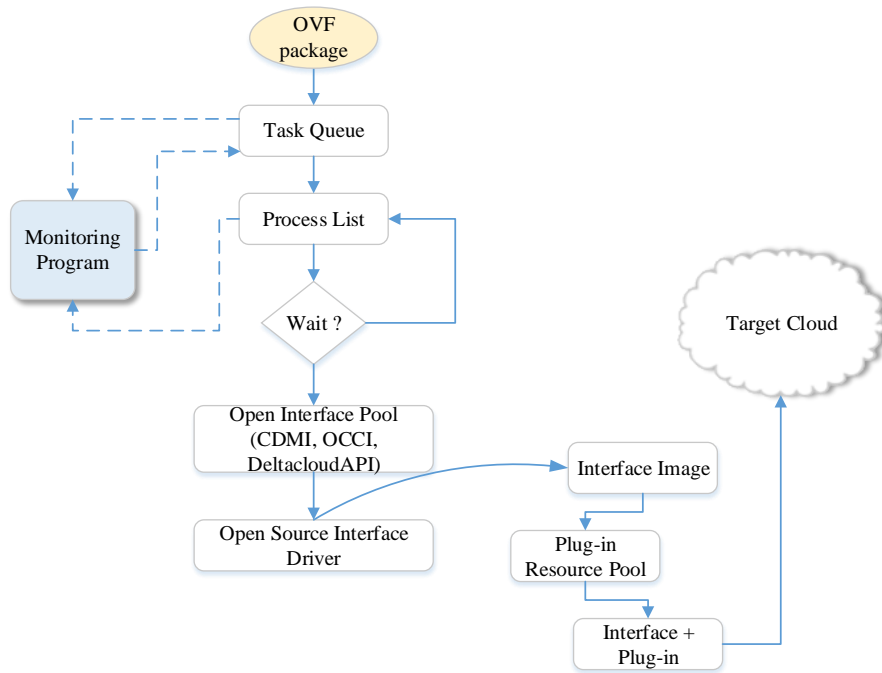


Figure 8. Matching Interfaces

OVF packages are sent to the task sequence, and ready to the matching interface procedure. Firstly, the task sequence accepts task requests, sends the latest task information to the monitoring program, and sends the task requests to the process list. Secondly, the process list determines whether run a new procedure, according to the current situation of the tasks, and sends the feedback information to the monitoring program. Then the monitoring program sends the feedback information to the task sequence. Thirdly, according to the demand, running processes select appropriate source interfaces in the open source interface pool and activate the interface driver, its operation can be seen in Figure 8. What's more, selected open source interfaces and drivers will be converted into image formats, and the client application will be started; then the cloud middleware selects the plug-in that matches the target cloud correctly in the plug-in resource pool. Finally, the cloud middleware packages the OVF packages by open source interface program and the plug-in of the target cloud, sending it to the target cloud.

5.3. Security of the Cloud Middleware

The cloud middleware plays an important role in the process of the workloads shift. Most importantly, its security is one key element of the entire service. Its security can be strengthened by the application of digital signatures and hash algorithm. We can see the process from Figure 9.

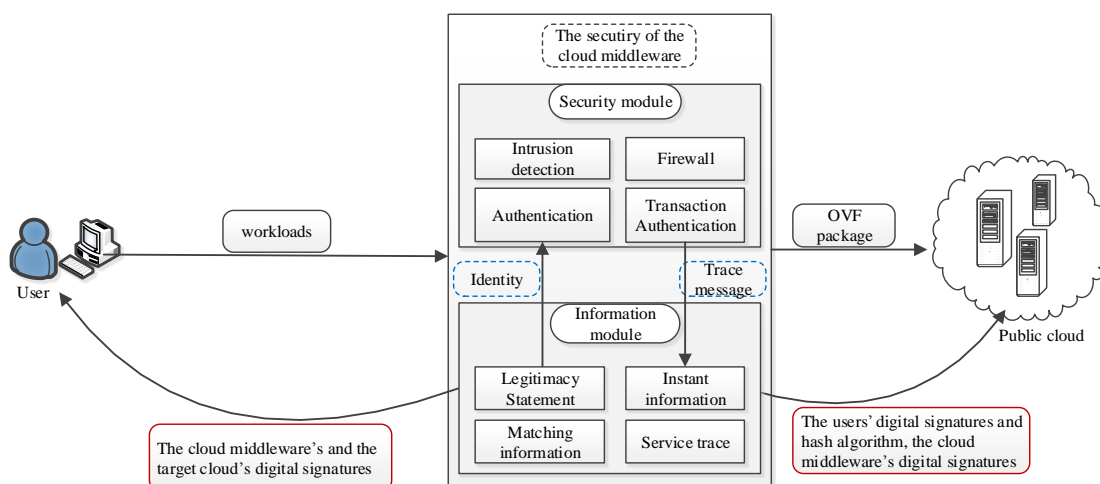


Figure 9. The Security of the Cloud Middleware

The cloud middleware is responsible for delivering the digital signatures and hash algorithm between users and the target cloud, as well as its own. The security module consists of authentication modules, firewall modules, intrusion detection modules, and transaction authentication modules. Service trace, instant messaging, matching message and legitimacy statement are the necessary parts of the message module. What is important is that the digital signatures can ensure the transaction integrity and the non-repudiation, and that the hash algorithm can ensure the integrity truth of the transaction.

6. Theoretical Verification

With the development of cloud technologies and standards, the issue that workloads cannot be easily deployed on different clouds will be gradually solved, moreover, with a strong theoretical and practical significance; the business model of the cloud service can also be more completed. In this program, based on the open source standards, it specifies

the role and technical details of the cloud middleware, and makes function modular, with a view to both the users' choices and the interests of suppliers.

7. Conclusion

With the further development of cloud service model, the service efficiency has been more and more important, and the cloud workloads deployment to the public cloud has been not only a hot point, but also an urgent problem, because of the technology monopoly between large-scale service vendors. However, the gradual improvement and development of the open source standards provides a great kind of tactic and idea to solve this problem. Against this background, this paper progress the construction program that is based on open source standards. This program is focus on the efficiency, security and expandability of the cloud workloads deployment, which owns the advantage of function modular, operation independence. What is more, it is contributed to improve the whole service efficiency, has a theoretical and practical significance. This program provides a guideline for the cloud workloads deployment. The refinement, optimization, verification of MapReduce algorithm, and the specific technical implementation of this program are the main research direction in the future.

Acknowledgements

This research was supported by the National Natural Science Foundation of China General Projects Grant No. 61272029, State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, China.

References

- [1] C Shuming, L Ning, S Zhexuan, X Xiang, "Introduction of Open Virtualization Format (OVF), Information technology and standare, (2012), pp. 32-42.
- [2] G Steffen, S René W. v, "App: A Standards-based Container for Cloud Providers", Operating Systems Review (ACM), 44, (2010), 115-123.
- [3] R Teckelmann, C Reich & A. Sulistio, "Mapping of cloud standards to the taxonomy of interoperability in iaas". In Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on IEEE, pp. 522-526, (2011).
- [4] S Wind, "Open Source Cloud Computing Management Platforms", 2011 IEEE Conference on Open Systems, ICOS, pp. 181-185, (2011).
- [5] J Matthews, T Garfinkel, C. Hoff, & J Wheeler, "Virtual machine contracts for datacenter and cloud computing environments". In Proceedings of the 1st workshop on Automated control for datacenters and clouds ACM (2009), pp. 25-30.
- [6] A. Edmonds, T. Metsch, A. Papaspyrou, & A. Richardson, "Toward an open cloud standard. Internet Computing", IEEE, vol. 16, no. 4, (2012), pp. 15-25.
- [7] A. Sampaio & N Mendonça, "Uni4Cloud: an approach based on open standards for deployment and management of multi-cloud applications". In Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing ACM (2011), (pp. 15-21).
- [8] De Chaves, S. A., Uriarte, R. B., & Westphall, C. B.. Toward an architecture for monitoring private clouds. Communications Magazine, IEEE, vol. 49, no. 12, (2011),pp. 130-137.
- [9] Karatas, F., Bourimi, M., Barth, T., Kesdogan, D., Giménez, R., Schwittek, W., & Planaguma, M.. Towards secure and at-runtime tailorable customer-driven public cloud deployment. In Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on, (2012),IEEE, , pp. 124-130.
- [10] C. A Mattmann, D Waliser, J Kim, P Ramirez., Goodale, C., Hart, A. F., & Cinquini, L, March). Earth Science informatics, 7, (2014), pp. 1-12.
- [11] L XiaoLing, W HuaiMin, D Bo, etc.. "MABP: an optimal resource allocation approachin data center networks", Science China-information Sciences, 57, 102801:1–102801:16 (2014).
- [12] A Sallam, Li, K., Ouyang, A., & Li, Z.. "Proactive workload management in dynamic virtualized environments". Journal of Computer and System Sciences, vol. 80, (2014), pp. 1504-1517.
- [13] Garg, S. K., Toosi, A. N., Gopalaiyengar, S. K., & Buyya, R.. SLA-based virtual machine management for heterogeneous workloads in a cloud datacenter. Journal of Network and Computer Applications, 45, (2014), pp. 108-120.

Authors



Xu Weixiang, He is presently a professor of School of Traffic and Transportation at Beijing Jiaotong University of China. He obtained his M.S degree in computer science from Dalian University of Technology of China in 1993, and PhD degree in System Engineering from Northern Jiaotong University of China in 2000. His current research interests focus on data mining, intelligent information processing and management, the analysis and integration for transport systems, and their joint applications in engineering design, Rail and urban rail transit for knowledge discovery and management purpose.



Jia Lin, She is a Master degree candidate of School of Traffic and Transportation at Beijing Jiaotong University of China. She received the B.S degree in e-commerce from Buohai University, Liaoning, in 2011. Her research interests include cloud computing and data mining.