

# Distributed Cloud Service Engine based on Edge Computing

Chen Zhong<sup>1, a</sup>, Xin Yuan<sup>2</sup>

<sup>1</sup>. Department of Computer Science and Technology, Tsinghua University, Beijing, 100083, China

<sup>2</sup>. Department IV of Foundation, North China Institute of Computing Technology, Beijing, 100083, China

<sup>a</sup> zhongchen0071@sina.com

**Abstract.** due to characteristics including multiple management, across localization, large user number and complex management, large-scale hierarchical organizations including governments and large enterprise groups, they require cloud platform architecture could adapt to a variety of flexible deployment patterns in the aspect of system construction deployment and adopt decentralized distributed computing architecture. Distributed cloud service engine adopts edge computing architecture to make cloud platform support graded system construction and management mode under multilevel data center environment, cross-domain access routing, resource scheduling, elastic scaling and information interaction problems under multistage deployment architecture, and apply to private cloud construction in the field with many deployment nodes and complicated node operating environment.

**Keywords:** edge computing; distributed cloud; multilevel deployment; industry private cloud; cloud service gateway.

## 1. Introduction

During cloud platform construction of government and large group enterprises, usually cloud platform can't comprehensively adopt centralized and flat architecture and provide various cloud service in the form of unified cloud data center due to some technical or non-technical constraints in network bandwidth, management mode and multi-business diversification. These industry sectors mainly construct multilevel distributed data center and cloud computing platform needs to adopt distributed architecture and support multistage deployment pattern. For this purpose, this paper raises distributed cloud service engine based on edge computing pattern to support deployment, operation and maintenance management of cloud platform under multilevel data center and distributed architecture.

External shielding distributed data center of distributed cloud service engine brings about cross-domain access routing, resource scheduling and information interaction problems to make multistage architecture of cloud platform be transparent to users, apply to multistage deployment system construction and application mode of organizations including party and government offices and group enterprises, and able to adapt to current situation of industry private cloud construction.

## 2. Distributed Cloud Platform

### 2.1 Multistage Deployment Architecture of Cloud Platform

Both government and large enterprise group belong to large-scale hierarchical organization, possess characteristics including multiple management, across localization, large user number and complex management, and require cloud platform architecture to adapt to a variety of flexible deployment patterns in the aspect of system construction deployment patterns that mainly include the following typical ones.

#### (1) Private cloud pattern of grand centralized deployment

Mode of subordinate unit access to application or service at headquarters under grand centralized pattern that system is centrally deployed at headquarters is shown in figure 1. This pattern is mainly based on wholly unified data center to establish private cloud platform and different units act as

tenants of cloud platform to use various cloud services in headquarters data center. Data center can adopt “two-place and multi-center” backup scheme and others.

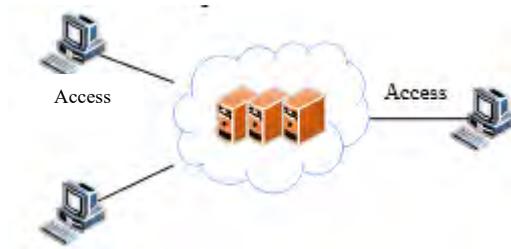


Figure 1. Grand centralized private cloud pattern

(2) Private cloud pattern of multiple network domain deployment

System deployment, management and maintenance of each subordinate unit are totally independent from those of headquarters and information interaction between headquarter and subordinate units is shown in figure 2. Headquarter and the subordinate units under this pattern are respectively based on independent data centers to establish private clouds that are only at the service of themselves. Information communication in system layer is realized through information engine and data sharing and exchanging mechanism.

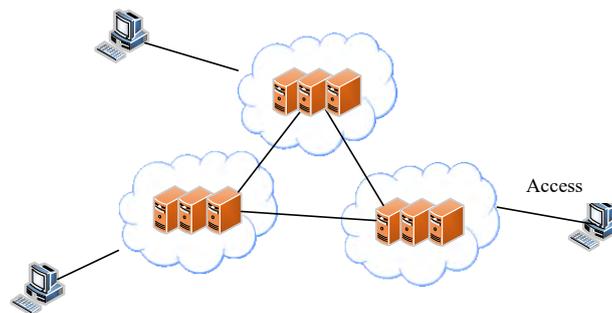


Figure 2. Multi-network domain private cloud pattern

(3) Distributed cloud pattern of multiple network domain deployment

Data centers of headquarters and subordinate units establish physically distributed and logically unified cloud through cloud platform in distributed architecture, as shown in figure 3. Main service that is nuclear, shareable and need unified requirement in business system is deployed on data center node at headquarters, each subordinate unit can visit or copy and share service or application at headquarters, application of subordinate unit itself is deployed on its own data center node. Unified access routing, resource scheduling and information interaction among different data center nodes are realized through distributed cloud platform.



Figure 3. Multi-network domain distributed cloud pattern

(4) Mixed pattern

Due to numerous complex factors including crew size of subordinate unit, management mode, multi-business diversification and network bandwidth, the several patterns above are mixed and adopted, which is the most common situation faced by system construction of party and government offices and large enterprise groups in reality.

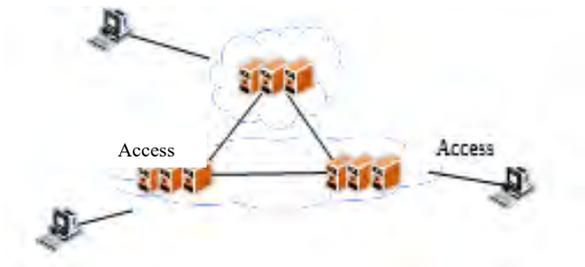


Figure 4. Mixed pattern

Thus it can be seen that distributed cloud platform needs to provide a technical architecture that supports multilevel data center deployment especially for the above four typical deployment patterns.

2.2 Distributed Cloud Platform Composed Architecture

Distributed cloud platform adopts edge computing concept, is based on multilevel distributed data center to provide business application system with unified cloud-based operation bearing environment, support service capabilities including fast application deployment as per request, elastic scaling, unified access to service, configuration management and log aggregation, and support serviceability replacement across information service nodes. Component architecture of distributed cloud platform is show in figure 5.

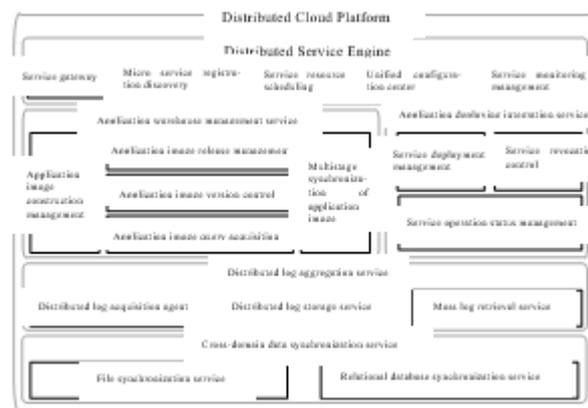


Figure 5. Component architecture of distributed service engine

Distributed service engine provides micro service deployed on distributed cloud platform with functions including service gateway, micro service registration discovery, service resource scheduling, unified configuration center and service monitoring management, supports unified access of micro service, request routing and access restriction, supports automatic log-on and discovery of micro service, and supports centralized management and coordination of configuration information for distributed service environment.

Distributed service engine is core component of cloud platform to support multilevel distributed deployment and is in charge of dispatching other components to complete distributed scheduling and access. Other components complete partial specific functions including data synchronization, log collection, rapid deployment and image management under the drive of distributed service engine.

### 3. Distributed Cloud Service Engine

#### 3.1 Cloud Service Access Gateway

Cloud service deployed by multilevel data center needs unified external access portal and unified access problem of services with the same name in different data centers can be shielded through cloud service access gateway, as shown in figure 6. Configuration data of cloud service access gateway is stored on different nodes of distributed cloud platform and data synchronization is conducted through real-time synchronization mechanism to guarantee mutual sharing of routing strategy among distributed cloud platforms so as to support the fact that different tenants can have access to any cloud service. Cloud service access gateway is independent from cloud service operating environment in the respect of design and supports distributed service request forwarding from the perspective of service access protocol.

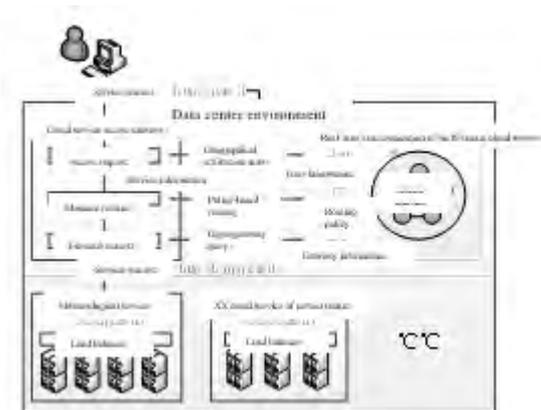


Figure 6. Cloud service access example

Cloud service access gateway is adopted to realize real-time synchronous access to routing data and it can support users' transparent access to cloud services deployed on distributed cloud platform, improve user experience, and promote overall reliability of cloud platform. Specific operating principle of cloud service access gateway is shown in figure 7. When request of cloud platform tenant reaches gateway in registration location, deployment location of actual cloud service requested by the tenant can be judged by querying tenant ownership information. When tenant and gateway have the same home location, tenant's request will be directly forwarded to the server of that service in registration location to complete request processing if there is no setting of other strategy; when home locations are different, tenant's request will be forwarded to the corresponding service of data center in user's home location for processing to realize user's cross-domain access.

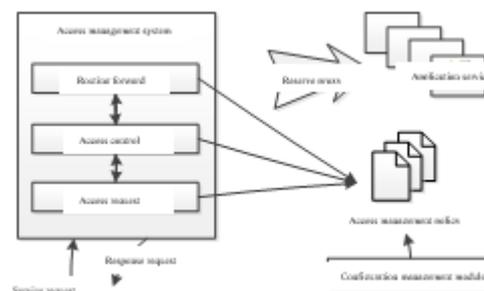


Figure 7. Operating principle of cloud service access gateway

As core components of distributed service engine, cloud service gateway is used to access external user or service request of the platform, conduct inward routing to load balancer at service end, and forward routing of service request according to pre-set rules. Its software function architecture is shown in figure 8.

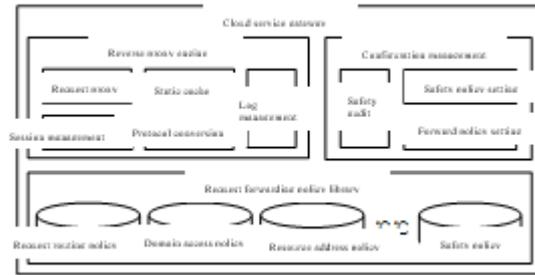


Figure 8. Functional architecture of cloud service gateway

### (1) Request forwarding policy library

Request forwarding policy library is the only basis service request of service gateway routing and is composed of a series of request forwarding policies and safety policy. Mainly path request policy, domain access policy, source address policy and safety policy are included. Path request policy guides service gateway to conduct routing forward according to URL path of service request, domain access policy guides service gateway to conduct routing forward according to host name of service request, and source address policy guides service gateway to conduct routing forward according to IP address of access requestor. In consideration of factors such as the existence of malicious attack on cloud platform, it's supported to add safety policy to guide service gateway to conduct deny of service towards some instable requests.

### (2) Reverse proxy engine

Reverse proxy engine is the core for service gateway to request routing forward and the forwarding basis comes from request forwarding policy library mainly composed of modules including request proxy, static cache, session management, protocol conversion and log management. Request proxy is mainly to complete proxy access to response service, session management is used to record requested session information, static cache is for caching service static resources to reduce pressure on service, protocol conversion is unpacking conversion for specific service protocol to reach reverse proxy effect, and log management is log recording for each reverse proxy request.

### (3) Configuration management

Configuration management is user oriented management platform of service gateway and is mainly used for setting relevant policies of service gateway. Mainly forwarding policy setting and safety policy setting are included. In the meantime, related audit work in the aspect of safety shall be arranged in the process of configuration management.

## 3.2 Unified Configuration Center

Distributed cloud platform usually brings about large difficulty to operation and maintenance personnel due to cumbersome configuration while conducting cloud service deployment in many data centers. Thus, it's necessary to conduct canonical modeling for various configuration parameters of basic cloud environment and realize unified storage management of configuration information on basis of data sharing synchronization.

Distributed cloud service engine maintains various configuration information through unified configuration library and provides functions including configuration template management, hierarchical configuration information synchronization, configuration visualization, configuration information subscription push and configuration retrieval. Unified configuration center, as unified configuration management platform for various resources on distributed cloud platform, is able to conduct centralized management application of configurations of different environments and different clusters and to realize real-time delivery to application client after configuration modification. Software function architecture is shown in figure 9.

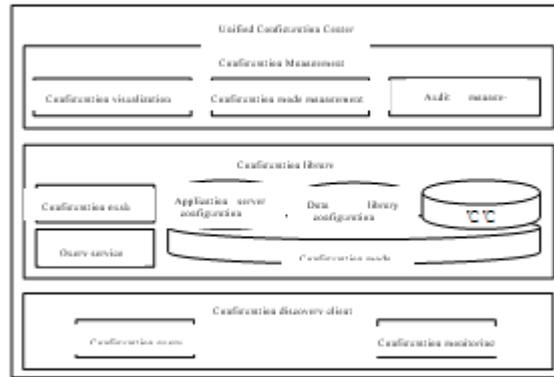


Figure 9. Functional architecture of unified configuration center

(1) Configuration management

Configuration management is user oriented management platform of unified configuration center and is mainly to provide users with functions including configuration visualization and configuration mode management. Besides, it's necessary for users to conduct audit operation for important configuration update in the process of configuration management.

(2) Configuration library

Configuration library is used to store core configuration parameters of unified configuration center. For example, parameters of service cluster usually include application server configuration and data library configuration. Besides, typical combined case system especially for various configuration parameters is stored as configuration mode.

(3) Configure discovery client

Configure discovery client is integrated in parameter finder system, provides parameter query interface, avoids direct interaction between parameter finder and configuration library, and reduces development complexity.

### 3.3 Distributed Cloud Resource Scheduling

Distributed cloud resource scheduling is a global resource manager of distributed service engine, provides unified resource framework, abstracts and divides computing, storage and network resources into various resource units from the infrastructure level, conducts state acquisition and scheduling management for the above resources, and can perform elastic scaling of cluster and hot standby for various business services according to resource demand and resource usage of business in control so as to guarantee normal operation of job tasks on cloud platform. Software function architecture is shown in figure 10.



Figure 10. Functional architecture of distributed cloud resource scheduling

(1) Service control

On one hand, service control provides dynamic control function for service cluster, supports dilatation and retraction of cluster node, and controls service cluster to conduct dynamic expansion adjustment after resource scheduling service generates corresponding resource scheduling policy according to the requirements on target performance of service cluster; on the other hand, service control provides hot standby function, automatically activates reserve service and launches reserve service instance at the same time so as to guarantee continuous operation of the service when main service goes wrong and fails to run.

(2) Resource scheduling engine

Resource scheduling module provides distribution and scheduling management of computing, storage and network resources of cloud environment. Resource scheduling service supports unified management of task scheduling framework, conducts resource scheduling and distribution in face of task, and provides resource alarm mechanism to support warning notice of other abnormal conditions such as lack of resources. Task scheduling plug-in module supports continuous task scheduling plug-in and self-defined task scheduling plug-in, and can conduct scheduling, distribution and automatic expansion of resources required by specific task, and execution control of task. Self-defined task scheduling refers to scheduling of extensible task such as timed task scheduling framework.

(3) Resource state acquisition agent

Resource state acquisition agent module is mainly to realize acquisition function of various resource states in run-time infrastructure. Acquisition of usage states of computing, storage, network and other infrastructure resources on cloud platform, acquisition of resource pool service on cloud platform server and software operation state, and acquisition of service and operation states of business in control are included.

### 3.4 Micro Service Registration Discovery

Micro service registration discovery guarantees mutual discovery among various service instances in distributed service engine and guarantees that other instances continue to provide services when instances in service cluster fail so as to guarantee high availability of the entire system. Software function architecture is shown in figure 11.

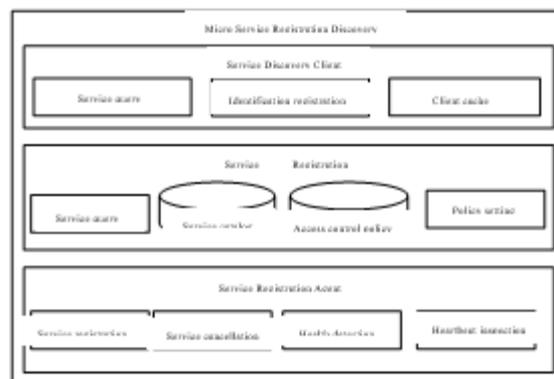


Figure 11. Functional architecture of micro service registration discovery

(1) Service discovery client

Service registration discovery client is integrated in service caller system, provides service call with service discovery interface, avoids interaction between service caller and service registration center, and reduces development complexity.

(2) Service registration center

Service registration center is core component of micro service registration discovery, provides client with service information query, and is mainly composed of modules including query service, service catalog, right control policy and policy setting.

(3) Service registration agent

Service registration agent is a tool for acquiring service instance and its information under distributed cloud platform environment, deployed on operation system of each host machine under cloud environment, and achieves automatic registration of service instance information through the event of creating monitoring container.

### 3.5 Cloud Service Monitoring Management

Cloud service monitoring management is the monitoring scheme that distributed service engine provides service and its operating environment, is based on service run log to analyze real-time status in the service operating process and to provide cloud service control with support, and possesses characteristics including malfunction alarm and flexible resource access. Software function architecture of cloud service monitoring management is shown in figure 12.

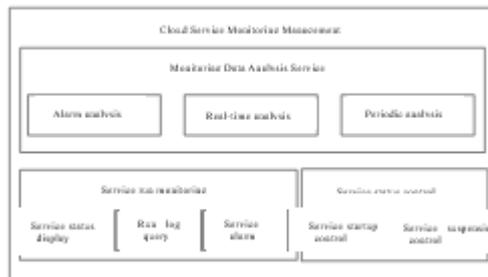


Figure 12. Functional architecture of cloud service monitoring management software

#### (1) Service running monitoring

Service running monitoring is mainly to provide view and browse of running states of various services on cloud platform. View of service running status, view of service running log and browse of service alarm information are included.

#### (2) Service status control

Service status control is mainly to provide control management of running states of various services on cloud platform. Service startup control and service suspension control are included.

#### (3) Monitoring data analysis service

Monitoring data analysis service is based on mass log data to conduct analysis on service run status. Log mining analysis is made to provide exception alarm interface for the convenience of integration with peripheral system, provide real-time data statistics function to support data acquisition of monitoring status display module, and provide periodic analysis function to conduct statistics of empirical data and guide reasonable scheduling and planning of the system.

## 4. Conclusion

Distributed cloud service engine conducts load balancing of high-concurrent requests of cloud platform, is able to conduct flexible and scalable resource scheduling for cloud services among multilevel deployment nodes, and can detect running states of nodes and make automatic allocation of resource refactoring nodes; in the meantime, it supports the system to adopt micro service architecture, is able to conduct health detection and rapid fault isolation for micro service instances deployed on the cloud, and can be used to realize distributed cloud platform design and apply to private cloud construction in the field with many deployment nodes and complicated node operating environment.

## Acknowledgements

This paper is funded by Special Domain Cloud Service and Integration Platform.

## References

- [1]. Cloud-Sea Computing Systems: Towards Thousand-Fold Improvement in Performance per Watt for the Coming Zettabyte Era[J]. Zhi-Wei Xu. *Journal of Computer Science and Technology* . 2014 (2).
- [2]. Internet of Things (IoT): A vision, architectural elements, and future directions[J]. Jayavardhana Gubbi,Rajkumar Buyya,Slaven Marusic,Marimuthu Palaniswami. *Future Generation Computer Systems*. 2013 (7).
- [3]. Service level agreement-based energy-efficient resource management in cloud data centers[J]. Yongqiang Gao,Haibing Guan,Zhengwei Qi,Tao Song,Fei Huan,Liang Liu. *Computers and Electrical Engineering*. 2013.
- [4]. Mobility First[J]. Dipankar Raychaudhuri, Kiran Nagaraja, Arun Venkataramani. *ACM SIGMOBILE Mobile Computing and Communications Review* . 2012 (3).
- [5]. Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing[J]. Anton Beloglazov,Jemal Abawajy,Rajkumar Buyya. *Future Generation Computer Systems*. 2011 (5).
- [6]. A view of cloud computing[J]. Michael Armbrust,Armando Fox,Rean Griffith,Anthony D. Joseph,Randy Katz,Andy Konwinski,Gunho Lee,David Patterson,Ariel Rabkin,Ion Stoica,Matei Zaharia. *Communications of the ACM*. 2010 (4).
- [7]. The cost of a cloud[J]. Albert Greenberg,James Hamilton,David A. Maltz,Parveen Patel. *ACM SIGCOMM Computer Communication Review* . 2008 (1).
- [8]. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility[J]. Rajkumar Buyya,Chee Shin Yeo,Srikumar Venugopal,James Broberg,Ivona Brandic. *Future Generation Computer Systems*. 2008 (6).