Chapter 20
A Cloud Computing Data Model

Hua-Zuo Ying

Abstract In this paper, an in-depth comparison is conducted on the architecture of data services, and also the shared disc architecture of the cloud computing database is analyzed. Besides, the application of the cloud computing is implemented based on the open-source cloud platform. Furthermore, the research result improves the parallelism and extensibility of the actual system, and simultaneously increases the data capacity.

Keywords Cloud computing · Cloud computing database · Data architecture

20.1 Introduction

The cloud computing, due to its key and revolutionary technology concept for IT, has penetrated into the hearts of people day by day and will exert a profound influence as well [1, 2]. A core design concept of cloud computing is designing a dynamic extensibility, or providing an on-demand service [3]. However, a majority of database services cannot fulfill such a demand at the present time. In this paper, the author discusses the implementation of a cloud database service after introducing the cloud computing database service architecture and the shared/non-shared disk architectures.

H.-Z. Ying (✉)
Center for Network Information and Computing, Beijing University of Aeronautics and Astronautics, Beijing, China
e-mail: zuoyinghua@hrsk.com

20.2 Cloud Computing

The cloud computing developed from the distributed computing, parallel computing, and grid computing; in other words, it can be a commercial implementation of these computer science concepts [4]. A large resource pool can be composed through the computation of the enormous and highly virtualized resources distributed in a great number of distributed computers. Besides, based on the open standard and service and centered at Internet, the cloud computing provides safe, fast, and easy data storage and network computing services.

20.3 Introduction to Cloud Database

20.3.1 Fundamental Concept of Cloud Database

The cloud database is a data management system based on the cloud computing platform, provides the parallel processing of mass data and good extensibility, and also can simplify the development of the cloud application. The application that the cloud database is appropriate to is mainly the OLAP application. The cloud computing databases are applications of cloud computing, and also are a cloud computing-oriented database resource management platforms, aiming to provide a cloud computing access environment for the sharing of the resources in many databases of Internet back ends and also provide a general plan for cloud computing applications to solve the fundamental structural-level database resources access, discovery, integration, and other problems.

20.3.2 Prominent Problems in the Cloud Computing Data Service

A core principle of cloud computing is the dynamic extensibility. The architectures of a majority of database system do not share the disks. The architectures not sharing disks rely on the distribution of data to different services. Perhaps, it is thought by people that the dynamic adding database is the same to the partitioning data. For example, if you get two servers, each of them will have 50 % data of a third server is added; also, there are lots of users to request to search relevant information; if they may want to look for the order information of all users, or may search an invoice list and all invoice information of last month, and then use relevant database keywords to search the customers list. If this is distributed in different data servers, the data shipping will smother the database performance. For this reason, the partition of database must be highly careful to reduce the data shipping. The partition of data, as a highly time-consuming process, is regarded as

a black art, because it requires a high-level technique. Certainly, the users can use the middleware to automatically partition data, but the performance will be reduced to be extremely low. Therefore, if a third server is added, it will greatly degrade the performance to use the middleware to dynamically partition data. This is the problem in the cloud computing database.

20.3.3 Data Replication Table Plan

The data partition does not match with the cloud computing, so Amazon and Google adopted plans to resolve the challenges of cloud computing. Hence, they created the persistent layer engine and gave up the typical ACID, and then adopted the data replication tables to support the dynamically elastic extensibility. Both the Bitable of Face book and the Sampled of Amazon are ideal for the above first application. However, they are not an alternative for the actual database, because they cannot satisfy the needs of clouding computing.

20.3.4 Shared Disk Data Architecture-Ideal Architecture of Cloud Computing

The shared disk architecture is unnecessary to partition data and is an ideal cloud database service. The shared disk database allows the low-cost cluster to use a single data set, and its typical application is SAN or NAS. All data can be accessed by all servers. There are no data partitions. As a result, adding more servers will improve the execution efficiency of statements. That is to say, the shared disk supports the flexible extensibility.

20.3.5 Database Solution Based on Cloud Computing

20.3.5.1 Eucalyptus Software Architecture

Eucalyptus is an open-source software architecture and implements the fundamental architecture of enterprises current IT service based on a cloud computing platform. The full name of Eucalyptus project is "Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems," which is an open-source project created by Santa Barbara University and is the software with the main purpose of fulfilling the elastic demands of the cloud computing environment. It is deployed on the clusters or server groups and also is a web-based service using commonly seen Linux tool.

20.3.5.2 Eucalyptus Plan System Structure

Eucalyptus platform framework is composed of a series of cooperative web services which are highly modularized [5]. The web services are interactive operation using the standard communication protocols. Through this framework, Eucalyptus implements virtual machines and storage resources which are connected by two isolation layers network. In terms of the client applications and users, the API is compatible with Amazon AWS (including SOA and REST interfaces support).

In Fig. 20.1, the CLC is the base (server, storage, and network) of the virtualized resources of cloud controller; the CCS cluster controller is the front end of each cluster definite in the cloud. The NCs is the machine to run the cases of virtual machines; the storage controller provides block services (similar to Amazon EBS), and the services can be extended to the whole cloud through Walrus storage system (similar to functions of Amazon S3).

In the deployment of the cluster servers, due to the shortage of IP address and the worry about the securities of the resources completely accessed through Internet, the system administrator usually deploys the cluster on a private and un-routable network, and this network has only one head node responsible for the routable traffic between the computing pool and the public Internet. Although such a configuration provides a security guarantee by using the least pubic and routable IP addresses, this means that a majority of machines cannot directly communicate with the machines within the cluster when connecting to the external hosts. Taking
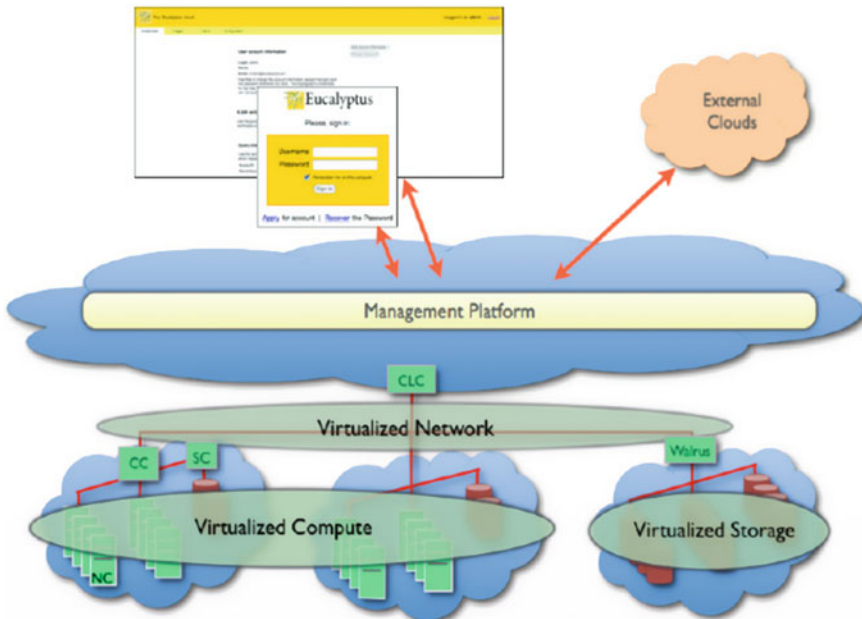

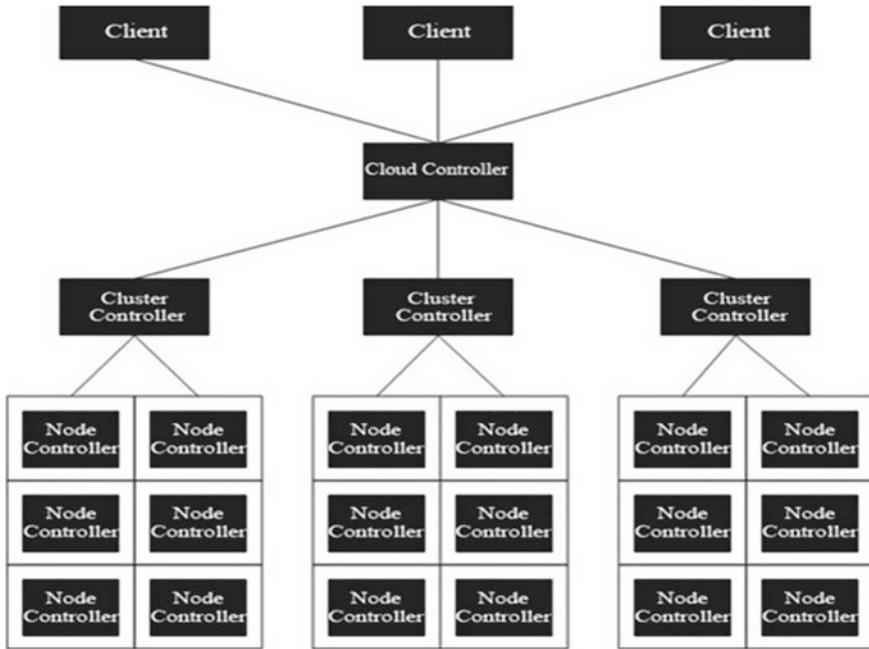**Fig. 20.1** Eucalyptus cloud concept

Fig. 20.2 Eucalyptus layered topological structure

the following circumstances for example, there are two small Linux clusters (a small server pool and workstation pool) and each cluster has a front-end machine whose IP address can be visited openly, while special networks are used to connect the spaces between nodes as well as the nodes and cluster head nodes; also the servers and workstations have open IP addresses, but these workstations are behind the firewalls and cannot be connected exteriorly; in this case, it is obvious that the installation of a completely interconnected system is impossible, because many machines can only send connections to the external hosts or are totally isolated from the external network. In addition, the nodes in the two clusters may have overlapped IP addresses because they are in different special networks. In order to use all these resources in a single cloud computing system, a layered architecture is applied in Eucalyptus as shown in the following Fig. 20.2.

20.3.5.3 Main Components

The main components of Eucalyptus include the node controller, cluster controller, and cloud controller.

20.3.5.4 The Node Controller

The node controller manages a physical node. It is a component running in the physical resources hosted by virtual machines and is responsible for the tasks of starting, checking, closing, and deleting virtual machines cases.

20.3.5.5 The Cluster Controller

The typical cluster controller runs in the cluster head node or server and can access the private or public networks.

20.3.5.6 The Cloud Controller

The cloud controller is composed of a group of services. These services are used to process the users requests, verification, maintaining system, and metadata (virtual machine mapping, SSH keys, and so on), and also can manage and monitor the running of virtual machine cases. These services are configured and managed through the enterprise service bus. Also, the operations such as service release can be implemented through the service bus. In the design of Eucalyptus, the transparency and easiness are stressed to promote the experiment and extension of Eucalyptus. In order to achieve the granularity-level extension, the components of the cloud controller include the virtual machine scheduler, SLA engine, user interface, and management interface. They are modularized components which are independent mutually and provide well-defined interfaces to the outside, while the enterprise service bus (ESB) is responsible for controlling and managing the interaction and organic cooperation between these services. By using the interactive operation of web services and Amazon EC2 query interface with EC2 client, the cloud controller can work as Amazon EC2. Why EC2 is selected? Is that it is relatively mature, has a great number of user groups, and has implemented Iasi excellent? As shown in Tables 20.1 and 20.2.

Table 20.1 The configuration requirements on other nodes

Hardware	Minimum	Suggested	Notes
CPU	1 GHz	2 × 2 GHz	For an all-in-one front end, it helps to have at least a dual core processor
Memory	2 GB	4 GB	The java web front end benefits from lots of available memory
Disk	5,400 rpm IDE	7,200 rpm SATA	Slower disks will work, but will yield much longer instance start-up times
Disk space	40 GB	200 GB	40 GB is enough space for only a single image, cache, etc. Eucalyptus does not like to run out of disk space
Networking	100 Mbps	1,000 Mbps	Machine images are hundreds of MB and need to be copied over the network to nodes

Table 20.2 The node controller (no)

Hardware	Minimum	Suggested	Notes
CPU	VT extensions	VT, 64-bit, multicourse	64-bit can run both i386 and amd64 instances; by default, eucalyptus will only run 1 VM per CPU core on a node
Memory	1 GB	4 GB	Additional memory means more and larger guests
Disk	5,400 rpm IDE	7,200 rpm SATA or SCSI	Eucalyptus nodes are disk-intensive; I/O wait will likely be the performance bottleneck
Disk space	40 GB	100 GB	Images will be cached locally, eucalyptus does not like to run out of disk space
Networking	100 Mbps	1,000 Mbps	Machine images are hundreds of MB and need to be copied over the network to nodes

Configuration Requirements:

The cloud controller (clack)

The cluster controller (cc)

Walrus (the S3-like storage service)

The storage controller (sc)

20.4 Implementation of Upgrade

On the Eucalyptus, cloud basis, huge virtual servers, and multiple virtual terminals can be created and implemented. Red hat Linux enterprise edition can be installed in the virtual machines and also the Oracle database can be installed in accordance with the shared disk architecture; also, the database in the former IBM4300 small machines can be transferred to virtual machines, and the former applications can be created in multiple virtual terminals; the application upgrades of the huge database can be implemented, to achieve the fast query of mass data and the flexible increase and decrease of application terminals.

20.5 Conclusion

The cloud database services are created through the Eucalyptus cloud platform. Such a database model service owns the features below:

Ultra-large scale: the HOSTING providing the cloud computing is more powerful than the common virtual hosts

East to use: the users can acquire the application services with multiple terminals at any position

High reliability: the data multiple replica fault tolerance, computing nodes isomorphism, and interchanges and other solutions are used to ensure the high reliability of services

Extensibility: the data and application sharing between different equipments can be achieved with this feature

However, there are two divergences in the IT world for whether the relational database can be appropriately used as the database in the cloud environment or not. The relationship between the Nasal in the research area and the traditional commercial database changes as well. The data services how to highly efficiently satisfy the multiple application models in the cloud environment are still necessary to be explored further.

References

1. Liu P (2009) Characteristics and definition of cloud computing. China Cloud Comput China Cloud, 18(4):112–116
2. Zhu J (2009) Smart cloud computing 42(11):87–93
3. Hogan M (2009) Cloud computing and databases 13(12):234–238
4. Denis (2009) A cloudy approach-SQL data services 4(34):556–562
5. Quan C, Ni DQ (2009) Cloud computing and its key technologies. J Comput Appl 5(9):562–567