

# NoSQL Distributed Big Data Storage Technology and Application Based on Cloud Platform

LU Zheng-Wu

Communication University of China, Beijing, China

e-mail: buaaspeed@gmail.com

**Keywords:** Hadoop cloud storage platform; MongoDB database; NoSQL; cloud storage.

**Abstract.** With the rapid development of mobile Internet cloud computing and big data, the data storage requirements, such as pictures, videos, etc., increase dramatically. At present, the existing storage methods can not meet the needs of system storage, and NoSQL distributed big data storage technology because of its scalable, fast read, mass processing and other characteristics, in the field of cloud computing has mushroomed widely used. The unstructured data cloud storage architecture was designed with MongoDB database technology based on Hadoop platform and NoSQL. The unstructured data storage technology of cloud storage can alleviate the difficulties existing in the current unstructured data storage, and improve the service quality of unstructured data storage.

## Introduction

With the development of Web 2.0 technology, more and more unstructured data (such as documents, pictures, music, video, etc.) generated. IDC studies show that 80% of current enterprises are unstructured data and are growing at a rate of 60%. How to break through the traditional technology, to achieve large-scale data efficient storage and use management is currently facing a major challenge.

## NoSQL database technology

Non-relational database (NoSQL) is a new non-relational distributed storage technology[10], the database data not only includes the daily text data, but also including pictures, videos, FLASH animation and other data, because of its massive storage, flexible Easy to use, high concurrency and other characteristics, the system can provide a scalable loosely coupled type data model, the model strictly follow the CAP theorem, can well support the storage of unstructured data to meet the needs of high concurrent read and write, Good scalability[3].

The current NoSQL data storage mainly includes the following, as shown in Table 1.

Table 1. NoSQL storage classification and features

| Type                    | Name                            | Feature   |
|-------------------------|---------------------------------|---|
| Key-value pairs storage | MemcacheDB, Dynamo, Redis       | Through the key index to quickly query to its corresponding value, saving query retrieval time  |
| Document storage        | CouchDB, MongoDB(Commonly used) | Generally used to store the basic attribute data, data encapsulation storage format similar to json string, the contents of the storage according to the characteristics of the document itself to facilitate the realization of some of the fields to index, so as to achieve the function of fuzzy search |
| Column storage          | Hbase, Hypertable               | Can facilitate the database storage of structured and semi-structured data, data write and read simple and quick, while easy to do data compression   |

(1) Key-value storage. Through the provision of key value on the data storage, can well meet the system read and write needs, such as MemcacheDB, Redis, Dynamo.

(2) document data storage. Can effectively meet the system's massive data storage and access needs, the current storage, including MongoDB and CouchDB, etc., can greatly improve the efficiency of mass data storage access.

(3) oriented column storage. In a column-oriented storage system, the data is listed as a storage unit. The same column of data is stored together to support the dynamic expansion of the column, for a column or a few columns of the query has a clear I / O advantage[12]. Typical representative of Hbase, Hypertable and so on[4].

## MongoDB database

The MongoDB database not only improves the fast write and read of the storage through the system's key-values, but also is well compatible with traditional relational DBMS functionality[5], which provides a powerful, flexible, and extensible unstructured storage Way[11], the storage method uses a loose BSON data structure for storage.

The main features of MongoDB are as follows:

(1) good scalability: MongoDB database in the early design, the first to consider the expansion of the database, through the use of model-free storage structure can be quickly on the server for automatic segmentation. Using MongoDB database automatic fragmentation mechanism, you can achieve the dynamic load balancing of the data in the cluster[6].

(2) Data storage rich and diverse: MongoDB is a document-oriented database, MongoDB abandon the relational storage model, the user to add modules without prior definition, you can change the data model flexibly and flexibly.

(3) Rich retrieve function: the database not only supports auxiliary index, and easy to store JavaScript script and MapReduce and other models.

(4) Excellent performance engine: MongoDB database in the document data can be dynamically expanded, usually the first storage engine configuration to the memory mapping file[5], the realization of data files for pre-allocation, and then through the use of space conversion form to ensure the stability of the database performance.

(5) Convenient configuration management: MongoDB database to enhance the reliability and stability of the system, usually using the server's own mechanism to complete the configuration function. The core of the MongoDB database is document data, and the field names and values in each document data are stored in the table. You can manage the corresponding database tables with the MongoDB database instance.

Figure 1 shows the MongoDB distributed storage architecture.

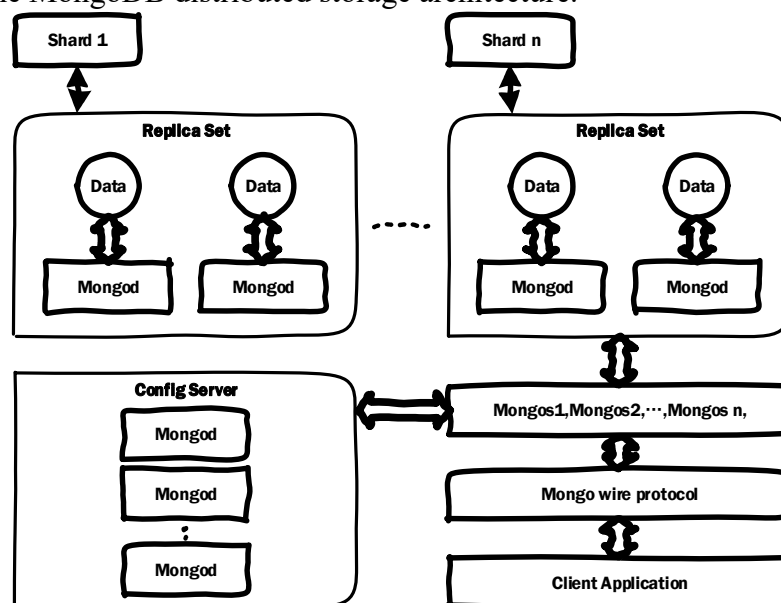


Figure 1 MongoDB distributed storage architecture

## Fragmentation

In the MongoDB database, each fragment consists of one or more servers, which function primarily to store the data by running the MongoDB process. But in the real world, in order to

improve the reliability of the system and achieve automatic fault recovery, each fragment can be independently seen as a replica set, replica set in essence, it is an asynchronous master-slave replication mechanism. A replica set includes at least one master node and one or more secondary nodes. The main node is mainly responsible for the data write, the deputy node is mainly responsible for data read. Replica set between the two types of nodes through oplog to ensure data consistency, all operational data and timestamps will be written to oplog, because of its size is fixed, all the sub-nodes will monitor the changes in oplog to achieve with the main node The synchronization. Replica set is usually able to achieve more than two sub-node failure automatic recovery[8].

### **Configuration services**

The configuration service is mainly used to store the metadata information of the MongoDB database cluster. The metadata information is derived from two aspects: one is the cluster information on the fragmented server; the other is the document data and the collection information on the fragmented cluster server. Each configuration service information includes the cluster information in the MongoDB database, which typically uses a two-phase protocol to communicate with each other to ensure consistency of configuration information. The configuration server currently has its own replication model that can back up cluster metadata information. When any cluster server is down, the metadata in the cluster becomes automatically read-only. In this way, the system can effectively prevent the system from being erroneously caused to cause the metadata information to be changed. There is a case where metadata is inconsistent between nodes. An exception in a configuration server in a database cluster does not affect the overall operation of the entire cluster, and ultimately ensures that data is written or read from the cluster.

### **Routing process**

Routing process can be a database cluster of multiple components as a single system, when the MongoDB database server to receive user requests, the first query the corresponding configuration file, find the data stored in the fragmented server. And then through the configuration service agreement to forward the user request to the corresponding fragment server. When all the fragmented servers have completed the operation, the results are sent to Mongos via the routing protocol package. When Mongos summarizes all the data results, the final result is returned to the user. Mongos every time you start, first go to the configuration server to read the metadata unit and save it locally. Whenever the metadata in the configuration server changes, it will notify all Mongos the first time.

## **Hadoop big data cloud storage platform**

The Hadoop Data Platform is an open source project for Apache, which is designed to generate a system-stable, scalable, distributed computing architecture based on current inexpensive hardware devices. HDFS[7] is one of the subprojects of the Hadoop platform[1]. It is mainly able to implement a distributed file system, which provides a reference for the construction of cloud storage solutions for major organizations and companies[2].

Cloud storage data center is composed of multi-server server cluster collectively, the cloud storage data center can provide large capacity, high concurrent data storage. The service cluster is mainly composed of a master node and multiple data nodes, which can connect the cluster system together through the network equipment to realize the unified management and maintenance of the user.

Unstructured data storage is very extensive, the current system construction function module requires a lot of unstructured data[9], including picture upload and download, news photo publishing, video clips, document management and other functions. The cloud storage hierarchy is shown in Figure 2.

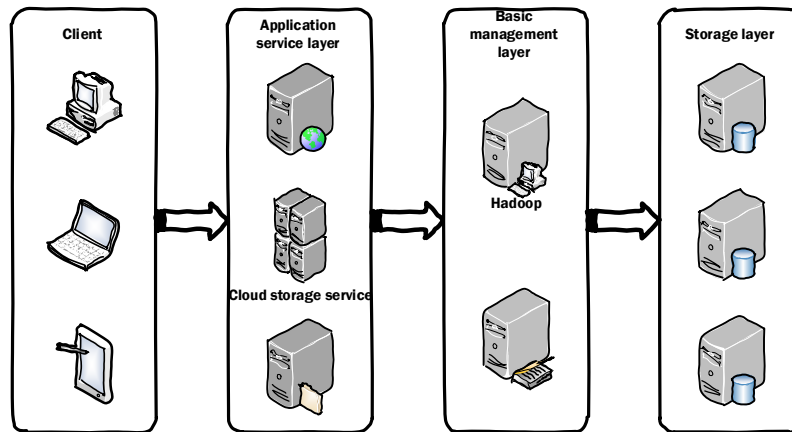


Figure 2 unstructured data cloud storage hierarchy

But most of the functional implementation is through the server to create a writable directory to store, using this approach has the following drawbacks:

(1) low performance. Because of the large amount of unstructured data required by the system, server bandwidth and computing power are allocated according to the data type, which results in a large number of server storage being occupied, which has a significant impact on servers with high performance requirements on some core devices.

(2) cluster synchronization is difficult to maintain. When large-scale project scale needs cluster support, in order to ensure that the synchronization between nodes within the server to achieve data, usually based on the agreement between the server to ensure the consistency and integrity of data transmission.

(3) service safety control poor. The current server are gradually using centralized control, which makes the server entry and exit security more stringent requirements, the traditional invasion is through the server upload Trojan to achieve, in order to better enhance the service security management, the urgent need for external service interface supervision.

(4) data security. For the exchange of server cluster data, usually using vpn login to obtain the required data information, and vpn is usually inside and outside the network information entry vouchers, vpn and other information to strengthen the supervision of information is to ensure data security is the most important.

(5) data persistence. Based on the business application of the system, usually a single database server storage, once the database exception, can not guarantee the integrity and consistency of data, prone to storage data loss, etc., once lost, data recovery is often more difficult.

The unstructured data cloud storage architecture is based on Hadoop, and the hierarchy is as follows:

(1) storage layer: unstructured data cloud storage system to provide different storage services, each service data is stored in the system, constitute a data pool.

(2) management: management is the unstructured data cloud storage system in the core layer. Through the management layer to achieve cloud storage in the interoperability between multiple storage devices to ensure that these devices to provide a unified public data services.

(3) Application service layer: This layer is mainly based on the user's own business and business needs, to achieve the cloud storage server cluster to interact with the specific operation, and ultimately the user's business operations.

(4) application interface layer: corresponding to the cloud storage user access layer.

### Improved HDFS storage design

In this paper, to optimize the HDFS storage, the introduction of MongoDB non-relational database corresponding improvements in the Hadoop cloud platform to build HDFS and MongoDB storage system to achieve the storage of unstructured resources. The unstructured data cloud storage platform is shown in Figure 3.

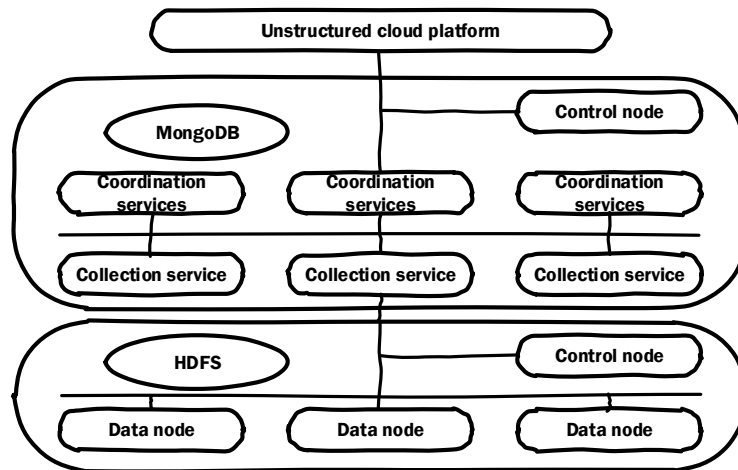


Figure 3 unstructured data cloud storage platform

Unstructured data cloud storage platform MongoDB architecture shown in Figure 4, its function is as follows:

- (1) Client: includes access to MongoDB interface, maintenance of cache data to speed up access to data, such as the location of the information.
- (2) coordination services: to ensure that the entire system cluster only one main control node, can store all the set of address entries, real-time monitoring of the status of the collection service, the collection service status information sent to the main control node; storage and management MongoDB Mode information, including which collections, each set of documents.
- (3) Control node: allocates aggregate space for load balancing of metadata storage; failover when a failed data node is found; handles junk file recovery and schema update requests on MongoDB.
- (4) Collection services: I / O requests for data; automatic fragmentation of collections with large amounts of data.

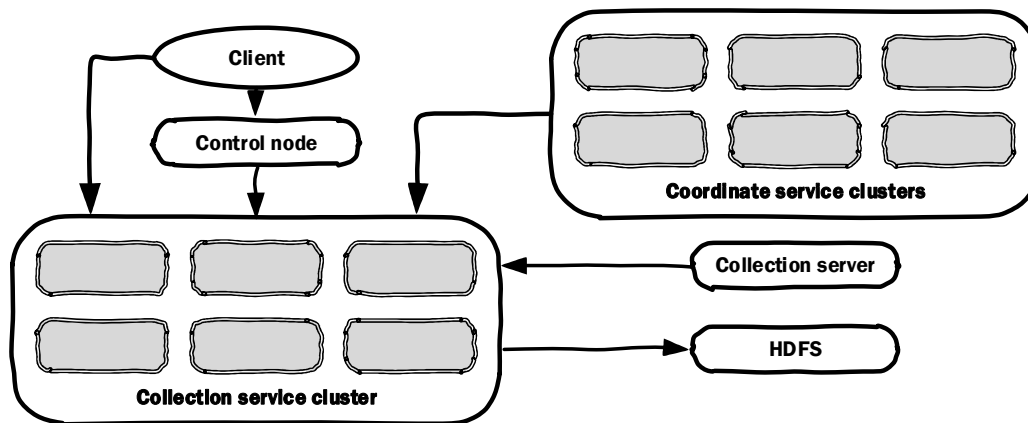


Figure 4 Architecture of MongoDB under unstructured data cloud storage platform

The unstructured data cloud storage platform HDFS architecture shown in Figure 5, its function is as follows:

- (1) NameNode node that is the control node, it can be seen as a manager in HDFS, the work content includes: management file system namespace, cluster configuration, storage block replication.
- (2) The storage node is the basic unit of data storage, and the data can be stored in the local file system in blocks, and all the block information can be sent to the control node in time and accurately.
- (3) MongoDB client is to obtain a distributed file system applications.

With the open source Hadoop unstructured data cloud storage platform, users can store massive unstructured data on the cloud platform. HDFS can be used to store PB-level data, but the data access delay is high, not suitable for storing small files. MongoDB storage of massive data of metadata, you can achieve real-time read and write, a good make up for the lack of HDFS.

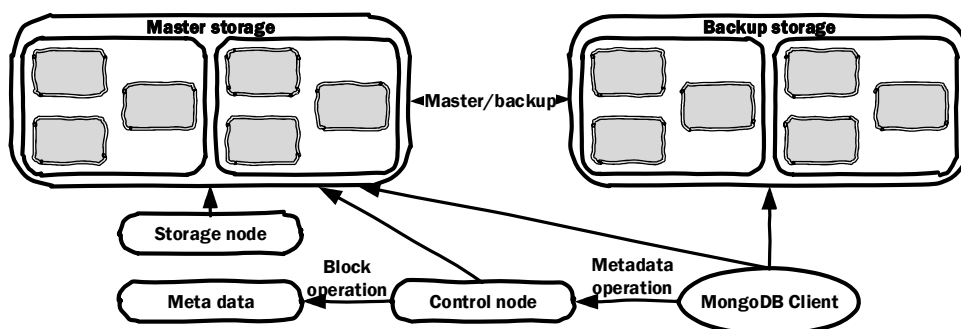


Figure 5 Structure of HDFS under unstructured data cloud storage

## test results

Test data from the company's business data. This experiment mainly on the Hadoop platform NameNode node memory loss test. In order to carry out the NameNode memory loss test experiment, the experimental design data are as follows: the number of samples is 10 000,20 000,30 000,40 000,50 000, each group of the number of samples to create 100,000 files; the introduction of MongoDB before and after the introduction of System to carry out the corresponding experiment, get NameNode node memory size and record, the experimental results shown in Table 2.

Table 2. Comparison of memory usage in NameNode

| Number of samples (PCS) | Original memory loss (Mb) | memory loss while Introducing MongoDB (Mb) |
|-------------------------|---------------------------|--|
| 10 000                  | 70                        | 60   |
| 20 000                  | 90                        | 62   |
| 30 000                  | 110                       | 64   |
| 40 000                  | 130                       | 63   |
| 50 000                  | 136                       | 65   |

As can be seen from the experimental data, for the original without any modification of the HDFS file system, with the increasing number of files, NameNode node memory consumption also showed a linear growth trend. After the introduction of MongoDB, the number of files is increased, the memory consumption of the NameNode node remains essentially unchanged, and the memory usage is not high. Table 3 is the response time comparison result.

Table 3. Response time comparison

| Number of samples (PCS) | Original response time (s) | Response time while Introducing MongoDB (s) |
|-------------------------|----------------------------|---|
| 10 000                  | 20                         | 20  |
| 20 000                  | 30                         | 22  |
| 30 000                  | 36                         | 21  |
| 40 000                  | 38                         | 24  |
| 50 000                  | 43                         | 22  |

As can be seen from Table 3, for the original without any modification of the HDFS file system, with the increasing number of files, the system response time linear growth trend. After the introduction of MongoDB, the number of files is increased, the system response time remains essentially unchanged. The experimental results show that, based on the Hadoop cloud storage platform, the combination of MongoDB and HDFS improves the memory consumption and response time of the big data NameNode node.



## Conclusion

Based on the analysis of the current situation of Internet development, this paper focuses on the current NoSQL database on behalf of MongoDB database model, characteristics and its distributed storage architecture, and details the characteristics of big data cloud platform technology, based on big data cloud storage Platform and unstructured database of its own characteristics, designed based on MongoDB and HDFS unstructured data cloud storage service architecture. For the characteristics of unstructured data metadata and the storage properties of MongoDB, MongoDB is deployed to the Hadoop platform to store metadata information for unstructured data, while unstructured data is stored in the HDFS system as a file. This reduces the performance bottlenecks caused by insufficient memory for NameNode when HDFS is responding to mass storage applications. At the end of this paper, the feasibility of the platform is verified, and the experimental data prove the effectiveness and efficiency of the improved scheme.

## ACKNOWLEDGMENT

This work is supported by Research project of National Development and Reform Commission (No. [2013]1309), Thanks to the reviewers for the valuable comments helping to improve the quality of the manuscript.

## References

- [1] WHITE Tom. Hadoop Authoritative Guide [M]. Beijing: Tsinghua University Press, 2011.
- [2] Li Qiao, Zheng Xiao. Study on the current situation of cloud computing [J]. Computer Science, 2011,38 (4): 32-37.
- [3] Xie Huacheng, Chen Xiangdong. Unstructured data access for cloud storage [J]. Journal of Computer Applications, 2012,32 (7): 1924-1928.
- [4] WANG R W, ZHANG H, DENG Y F, et al. Efficient parallel radiosity for terascale applications [C] // Proceedings of 2008 International Conference on Computer Science and Software Engineering. [S.l.]: IEEE, 2008: 1074-1077.
- [5] FABIANOWSKI B, DINGLIANA J. Interactive global photon mapping [J]. Computer graphics forum, 2009,28 (4): 1151-1159.
- [6] TIAN Lang-jun, CHEN Wei-wei, CHEN Wei-dong, et al. Study on Dynamic Load Balancing Algorithm in Cloud Storage System [J]. Computer Engineering, 2013,39 (10): 19-23.
- [7] GHEMAWAT S, GOBIOFF H, LEUNG S T. The Google file system [C] // Proceedings of 2003 ACM Symposium on Operating Systems Review. [S. l.]: ACM, 2003: 29-43.
- [8] LIN Fei, ZHANG Wan-jun, SUN Yong. A distributed unstructured data replica management model [J]. Computer Engineering, 2013,39 (4): 36 -38.
- [9] Hu Shanshan. Research and application of unstructured data storage for cloud storage [D]. Guangzhou: Guangdong University of Technology, 2014.
- [10] Li Cunchen. Mass data distributed storage technology research and application [D]. Beijing: Beijing University of Posts and Telecommunications, 2013.
- [11] Yang Lei. Based on NoSQL database of structured storage design and application [J]. Science and Technology, 2011 (18): 99.
- [12] Zhu Jiansheng, Wang Jianxiong, Zhang Junfeng. Study and application of big data query technology based on NoSQL database [J]. China Railway Science, 2014, 35 (1): 135-141.