Study on HDFS improvement scheme based on the GE code and dynamic replication strategy

Song Fei*, Cui Zhe

Chengdu Computer Institute of Chinese Academy of Science

Received 1 March 2014, www.tsi.lv

Abstract

There is a lot of valuable information in the massive amounts of data. Any loss of data may result in a great loss. Data security cannot be ignored. There are varieties of data disaster recovery technologies. However, most of these techniques depend on the hardware devices or data redundancy greatly. This paper presents a distributed data disaster recovery technology that minimum dependence on data redundancy and hardware system redundancy. In addition, this technology has nothing to do with the user equipment and application data structures. The test proved that this new data disaster recovery method can not only enhance disaster recovery capabilities and reduce the redundancy of the system greatly, but also suitable for large-scale distributed data disaster recovery.

Keywords: data disaster tolerance, HDFS, GE code, dynamic replication

1 Introduction

Now the data disaster recovery technology is divided into two categories: the first-class technology is closely related to the hardware devices and data structures. The first-class technology focused on the probability of damage to data equipment and the solution is based on a new type of memory devices and integrated device instead of the old data devices. The second type of study continued the system redundant technology roadmap (equipment redundancy and data redundancy). The second type of study is not very high to the quality of the storage devices, but the realization cost is closely related to the goal of the data disaster recovery system [1]. The lower target disaster recovery system only requires that the system can be able to provide the most basic data and services when the data was accidental damage. The higher goal of disaster recovery system requires no matter how much the price, the system can be able to recovery the lost data and interrupt service completely [2]. Because these two types of technical are exist the problems of higher redundancy for equipment and data, so, this paper presents a data disaster recovery method that based on encoding and dynamic replication strategy [1].

This new method based on computer reasoning [3] and data security storage [4] and it is a distributed technical method of disaster recovery storage [5]. This method is help to expand the scale of the disaster recovery system and enhance disaster recovery capabilities and greatly reduce system redundancy data. The disaster recovery capabilities of the new method mainly depend on the organization of data and the support of software method. It has a minimum dependence on data redundancy and hardware system redundancy and has nothing to do with the user equipment and application data structures. This method is equally effective for the data disaster recovery that at the network distributed system-level and the level of memory storage units, and the larger the scale of the disaster recovery system, the better the effectiveness of disaster recovery. This technology is not a copy strategy of the data and it does not distinguish between work equipment and backup equipment, but with a minimum of redundancy cost to achieve the high efficiency of data disaster recovery and data recovery. Once a large area data devices disaster, this technology can be taken over the services and completed data recovery in a relatively short period of time and the cost of doing so just only software execution time of during data recovery. Therefore, this technology reduces the cost of storage space and equipment redundancy costs greatly [2].

2 GE Code

2.1 GE CODE TECHNOLOGY

Based on the low redundancy and high-performance data disaster recovery technology, this paper proposed a new type of array erasure coding technique that is E code family (including AE code and GE code). It expanded the data recovery performance and its range of applications for traditional array erasure codes [6-7]. The E code family has the significant advantages of powerful erasure and erasure parameters unlimited. Its operation is completely established bit arithmetic in finite field GF (2) and thus avoiding the difficulty of encoding is performed on a large-scale finite field [3].

In the E code, the information will be stored in an n × n data array. There are n blocks in a data array. Each data block is divided into n segments. Using segments n-t, …,

**Corresponding author* e-mail: asfei@aliyun.com

n-2, n-1 as the check segment and the rest k=n-t segments as the information segments.

A specific n=16, k=12, t=4 data array instance shown in Figure 1.

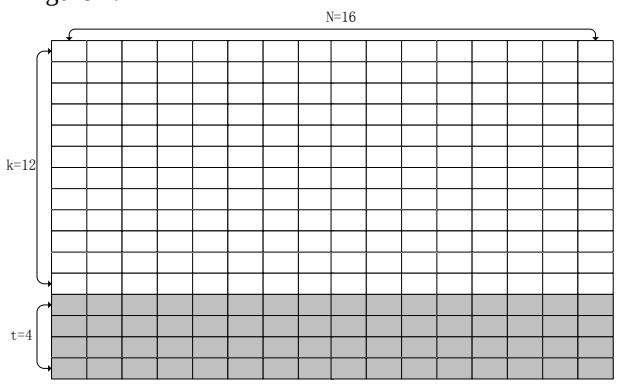


FIGURE 1 E code data array layout

As shown in Figure 1, E code is a parameter array code [n, k (n - k) / 2 +1]. This means that the length of E code is n and it can be able to accommodate k columns effective information and then we can use t = n − k columns of parity information to get columns of fault tolerance ability.

Using U represents a storage unit and using n as the number of storage units. We agreed and damaged number of the storage units is here.

In order to clearly stated, we agreed that t/n=q%. A typical case is q=25, then t/n=q%=1/4.

GE code process is as follows.

1) There are n numbers of U and each U is divided into n blocks average. The number of y^{th} block in the x^{th} U is a_{yx} and $0 \le x \le n$, $0 \le y \le n$. The before n-t blocks are blocks of information and that are used to store the effective information. After t blocks are parity blocks and which are used to store the parity information.
2) Construct a [n-t, n-1] binary matrix and homogenizing the matrix so that the difference of the maximum line weight and the minimum row weight is less than 1, the difference of maximum column weight and minimum column is less than 1 also. Each line of the matrix has n 1 and each column has j 1. All the elements of the value of 1 in the coding matrix obtained is saved as a collection of tuples A.
3) Let a_{ij} as the element in the data array on j-th column i-th row, we can use matrix $A=[a_{ij}]$, $0 \le i \le n, 0 \le j \le n$ to express E code. In matrix A, the information elements with $[a_{ij}]0 \le i \le n-t, 0 \le j \le n$ indicated and the parity elements indicated by $[a_{ij}]n-t \le i \le n, 0 \le j \le n$.
4) These elements randomly divided into t collections, as follows:

$D_0 : [d_{0,0}, d_{0,1}, \cdots, d_{0,n-t-1}]$

$D_1 : [d_{1,0}, d_{1,1}, \cdots, d_{1,n-t-1}]$

……

$D_{t-1} : [d_{t-1,0}, d_{t-1,1}, \cdots, d_{t-1,n-t-1}]$

So, the parity elements can be generated by the formula $a_{i,j} = \bigoplus_{s=0}^{n-k-1} d_{i-(n-k),s}$ $n\text{-}k \le i \le$ n-1 , $0 \le j \le$ n-1 .

5) When a U goes wrong and need to restore the data stored within the U, we first need to do is marked the status of all the parity blocks in U that no error has occurred as "available".
6) Choose an "available" status parity block randomly and check whether the information block that verified by parity block is deleted. If there is no information block is removed, or only one information block is deleted, then the parity block is marked as "useless". The deleted information block can be restore according to the parity block and the XOR operation results of other information blocks which verified by the parity block. A recovery formula is: $d_{i',j'} = a_{i,j} \oplus (\bigoplus_{s=0}^{n-k-1} d_{i-(n-k),s})$ i-(n-k) ≠ i',s ≠ j'
7) Repeat step 6, until all of the U, are no longer contains the wrong information blocks. At this time, all of the data stored in the broken U have restored.

2.2 DISASTER RECOVERY PROCESS

Suppose there are n storage devices, either t piece of equipment fails or is unable to obtain its data. Let q%=t/n. If the system is able to fully recover data within the device failure t, the system can be considered a successful disaster recovery. At this time, the effective storage space of the system is not less than (1-2*q %) times of the total amount of n storage device storage space, usually (1-q %) times. If set q=25, then the effective storage space is n devices storage space total quantity 75% and the rest of the space for additional coding information.

Stored procedure as follow.

1) The application procedure will line up source files according to the FIFO order when the application receives the file storage request and then the source files will be deal with one by one. The application procedure will segmentation and coding each source file accordance with the GE coding scheme. Each source file is divided into n sub-file average and the bit lengths of each sub-file fragment are same.
2) The application procedure sent the n sub-file of the source file f separately to the different n servers on the network storage system accordance with the pre-set parameters n. The total amount of sub-files on n servers of the network storage system is less time than the amount of data in the source files (1.5 times the total amount of source data typically). At this time the stored procedure is completed.

Download and read process in the conventional case as follow.

1) When the application procedure receives a download request to read a source file f, it sends to a download request to the remaining n-1 servers in the network system and collects all the transferred sub-file fragmentations.

2) After all n fragments collected from the source file f (one local server fragment, n-1 offsite server fragments), local application directly recovery data of source file f from n debris and thereby completing the download/read of the source file f. Because the sub-file fragmentation is complete, so this conventional process read from the source file f without decoding time loss.

Download and reading process in exceptional circumstances (Disaster recovery mode) as follow.

Assuming that there are certain units (the number t) of data storage servers in the network system failure or unable to respond.

1) The application procedure will send a handshake information broadcast to other n-1 servers in the network system after it gotten the download/read request from the source file f. There are at least undamaged servers will return the handshake information.
2) Chooses servers stochastically from the undamaged servers as the partner servers of data recovery. In accordance with step 2, the source file f can be recovered and reconstructed.

Attention. There is no need to recover each sub-file fragmentation of source file f and not need to re-coding f too. We can complete the data stored profile of source file f just need to recovery the t error fragments and restore them to the t intact servers.

Different source files are independent of each other and the reconfigurable architecture recover decoding is independent of each other too. Therefore, if there are multiple files need to restoration and reconfiguration, we can assign these files to faultless servers and carry out restoration and reconfiguration using the mode of distributed and parallel processing. This method can speed up the speed of data recovery and improve the efficiency of the disaster recovery system [4].

3 HDFS improvement program based on coding and dynamic replication strategy

This paper presents a new and improved solution based on GE code and dynamic replication strategy - Noah (Not Only A Hadoop). This solution based on Hadoop platform architecture and improvements to the underlying file system of HDFS and to achieve a low redundancy and high-performance disaster recovery for the huge amounts of data.

3.1 HDFS EXISTING MULTI-COPY STRATEGIC ANALYSIS

The operation mode of HDFS cluster is “master-slave”. The cluster contains two main types of nodes: Namenode (master) and Datanode (slave). Namenode can only have one, but Datanode can have a plurality. The mode of data disaster recovery of the existing HDFS is the multi-copy technical. The specific approach is saving the three copies of the file blocks on the different DataNode of HDFS cluster respectively. The NameNode responsible for complete copy work, it uses regular round-robin fashion to receive the heartbeat of each DataNode in the cluster. If there has the heartbeat signal, which means that the DataNode is working, conversely indicates DataNode is not working properly. If the NameNode cannot receive a DataNode heartbeat, NameNode will release the blocks of DataNode to the other nodes. NameNode use this method to keep the number of copies [5].

The main role of multiple-copy in HDFS as follow.

1) Fault tolerance: to ensure system reliability.
2) Load balancing: The size of the DataNode load is determined by how much of the data it has.

Therefore, HDFS average distributes the data to each DataNode to achieve the load balancing. During the running process, HDFS may transfer the load by the way of moving copies.

3.2 THE DESIGN IDEAS FOR NOAH

In summary, the main reason for HDFS system has higher storage cost and lower load balancing capability is multi-copy strategy with a fixed number of copies. Therefore, this paper tries to provide a more flexible load balancing solution, which not only ensure data security but also reduce the cost of storage. This program design ideas is based on (1). Use code-based disaster recovery technology to replace the multiple-copy disaster recovery (2). Abandon the old fixed number of copy strategies, using GE code-based disaster recovery mechanism to realize the strategy that change the number of copies dynamically [6].

4 Noah disaster recovery technology realizations

Noah disaster recovery program abandoned HDFS mirror copy policy and using coding fragment solutions to re-encoded the data blocks in HDFS. This way makes the whole system cluster save only the code section that corresponding with the data block. Although the mapping table and the name space is maintained in the Namenode, but the actual data storage is based on code section and the data read and disaster recovery is also in the Namenode which operation on the code section that corresponding to the data block. Noah using dynamic replication strategy instead of a fixed copy of the original HDFS strategy makes the whole system to keep the load balancing in a good state. The architecture of Noah is shown in Figure 2 [7].

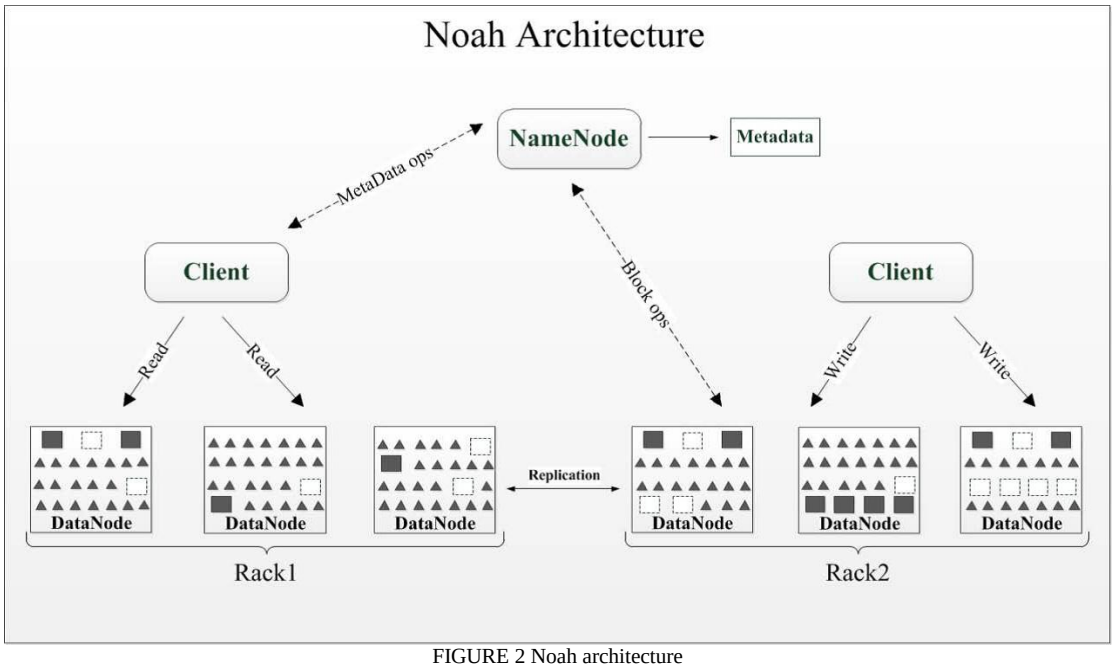


FIGURE 2 Noah architecture

4.1 DYNAMIC REPLICATION STRATEGY

According to the different needs of users for data and the system in support of the original the Hadoop file management mechanism, the dynamic replication strategy can make the system to maintain the load balancing in a good state by dynamic generation, delete and deploy the copies. For the hot data, dynamic replication strategy consists of the following two parts:

4.1.1 Dynamic replication generation strategy

Dynamic replica generation strategy has changed the way of saving a fixed number copies, it dynamically change the number of copies to achieve overall system load balancing. It is based on the needs of runtime system to decide the number of the copies for each data blocks. When generating copies, NameNode will be maintenance the number of the access-waiting queue for each data block that in the upper application. The maintenance principles are:

1) If the number of copies of a data block can be meet the application requests, the application requests do not need to wait in the queue, it can access to the copies directly.
2) If the number of copies of a data block can not be meet the application requests, the new application requests are entered into the data block's waiting queue, the system will generating several new copies for the application requests.
3) If the number of copies of a data block far beyond the number of application requests, it will produce a larger system consumes and will lead to a serious waste of storage resources. So, in order to improve utilization of storage resources, the system will delete the extra copies.

4.1.2 Dynamic copies arrangement

As mentioned earlier, the heartbeat is used to detect a DataNode working properly. Therefore, the DataNode notification the NameNode the system resource loads using the characteristic of the heartbeat. When the NameNode received the information from DataNode, the NameNode will judge the situation for each DataNode, and then the NameNode will move the higher load data block into the lower load data block and record the new position information of the transferred data blocks. Therefore, dynamic copies arrangement strategy is using this load transfer method to achieve a system load balance [8].

4.2 DATA RECOVERY STRATEGIES

The data recovery strategy of HDFS is replica the replication. This program presents a method that collect and decoding the section to achieve damaged data block recovering.

The section collection complete by the NameNode. NameNode first found the section that belongs damaged data blocks and then arbitrarily choose 70% of these sections put into a DataNode to decoding and data recovery operations. In the data recovery process, there will appear two situations:

a) Code section is lost or damaged. In this case, NameNode will record the information of the coding segment until the quantity of missing or corrupted code section exceeds a threshold (generally 30%). NameNode will notify the coding segment locations to the DataNode that own these coding segments. DataNode will decoding and parsing these coding segments and then distribute these re-encoded code section to different DataNode. So as to ensure the

number of code section maintained at a safe threshold value [9].

b) When the data of a hot spot data blocks are damaged or lost, NameNode will find the coding segments that corresponding this hot spot data blocks and then sent the coding segments to the specified DataNode. The DataNode decoding of these coding segments and thereby restoring damaged or lost data blocks, ultimately achieve the data recovery.

5 Experiments and conclusions

Experiments goal. For a distributed data storage system that consisted by N (3 <N <1200) data equipment (Such as memory cells, memory or a server), we need to check that when any of r data devices have corrupted (r<N/4) , whether the remaining N – r undamaged data devices can recover all the data and if the total redundancy of system equipment is less than 1:1.

The experimental results. In this study, we used 35 sets of equipment as servers and according to GE code divide the data into 35 parts. Each part of data size ranging from 1M to 600M. Then we imitate the data corruption status, shut down seven sets arbitrary and using the remaining 28 sets of equipment to restore the corrupted data.

The experimental results showed that: when the storage system of any r (r <N / 4) data device was damaged, all data can be immediately and automatically recover by the remaining N – r undamaged data devices and the total redundancy of the system storage equipment less than 1:1.

Figure 3 is a data disaster recovery resource consumption comparison chart for HDFS and Noah.

From figure 3, we can see that compared with HDFS 1:1 backup replication, Noah saves about 30% of the resources. And we can see the larger the backed up data the greater the Noah saved resources. Therefore, we can infer that Noah is more suitable for large-scale distributed data disaster recovery.

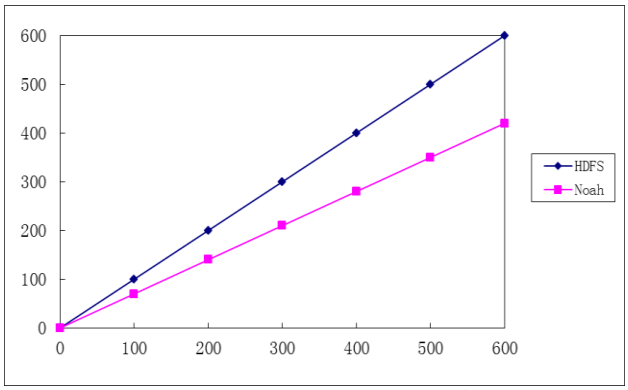


FIGURE 3 Resource occupancy comparison chart for Data disaster recover

Figure 4 and Figure 5 provide the compare for HDFS node load and Noah load node.

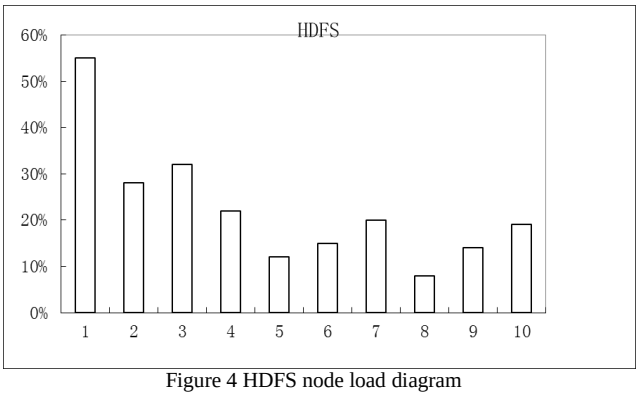


Figure 4 HDFS node load diagram

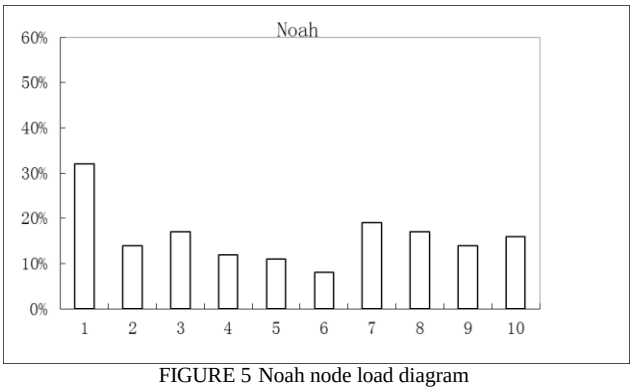


FIGURE 5 Noah node load diagram

From Figure 4 and Figure 5 we can see that the original HDFS load extremely uneven and fluctuation is larger, but Noah use dynamic replication strategy and allocates the load of a single node to each node in entire server cluster. and ultimately causing each node resource utilization more evenly and achieve load balancing.

6 Conclusions

The improvement program that proposed by this paper relative to original HDFS file storage system has following advantages.

This program lead encoding technology into HDFS file system and use an innovative way to replacing the original multi-copy disaster recovery technology. Original HDFS file system needs 1:1 redundant space, but this program just need no more than 70% redundant space.

This program adopts dynamic replication strategy and flexibility to change the number of copies replaces a fixed number of copies. It provides a more efficient system server load balancing ability and makes the distribution of resources of the entire file system more reasonable, more rapid and smooth running.

Because this program using the coding techniques, the users need not retrieve all the data when they requesting data and they can using the decoding method to restore the full data. When the parts nodes in the storage server failure or network congestion, this program has higher practicality.

References

[1] Benzmueller Ch, Sorge V, Jamnik M, Kerber M2008 *Journal of Applied Logic* **6**(3) 318–42

[2] Shuang K, Wang C, Su S 2010 A Novel Disaster Recovery Strategy in NGN core network based on P2P Technologies *Computational and Information Sciences* 601-4

[3] Fei Song, Wang Xiao-Jing, Zhe Cui 2012 A trust model of P2P in cloud computing environment *Advances in Mechatronics and Control Engineering* 1962-5

[4] Li Mingqing, Shu Jiwu, Zheng Weimin 2009 GRID Codes: Strip-Based Erasure Codes with High Fault Tolerance for Storage Systems *ACM Transactions on Storage* **4**(4) 15

[5] Dimakis A G 2011 A survey on network codes for distributed storage *Proceedings of the IEEE 99.3* 476-89

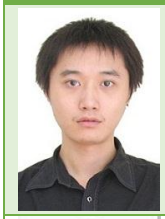
[6] Porter G 2010 *ACM SIGOPS Operating Systems Review archive* **44**(2) 41-6

[7] Sethia P, Karlapalem Kr 2011 *Engineering Applications of Artificial Intelligence* **24**(7) 1120–7

[8] Kambatla K, Pathak A, Pucha H 2009 Towards optimizing hadoop provisioning in the cloud *Proc of the First Workshop on Hot Topics in Cloud Computing* 118

[9] Shvachko K, Kuang H, Radia S 2010 The hadoop distributed file system *Mass Storage Systems and Technologies* 1-10

Authors



Fei Song, born in October 31, 1982,Chengdu,Sichuan,China

Current position, grades: Doctor studies
University studies: Chengdu Computer Institute of Chinese Academy of Science
Scientific interest: Data Disaster Recovery, Reliability Engineering and Network coding
Publications: 4
Experience: Song Fei is a doctorial student in Chengdu Institute of Computer Application, Chinese Academy of Sciences, China, has published more than four articles in reputed international journals and International Conferences, research interests are in the areas of Data Disaster Recovery, Reliability Engineering and Network coding.

Zhe Cui, born in September 20, 1970,Chengdu,Sichuan,China

Current position, grades: Professor
University studies: Chengdu Computer Institute of Chinese Academy of Science
Scientific interest: Trusted Computing, Embedded Systems and Reliability Engineering
Publications :15
Experience: Cui Zhe is currently a professor in Chengdu Institute of Computer Application, Chinese Academy of Sciences, China. His research interests lie in Trusted Computing, Embedded Systems and Reliability Engineering