A research on multi-level Networked RAID Based On Cluster Architecture[1]

Liu Xiao-Guang , Wang Gang and Liu Jing
Department of CS, Nankai University, Tianjin, 300071
E-mail: {lxgmail, wang_gang}@eyou.com

Abstract Storage networks is popular solution to constraint of server in storage field. As described by the Gibsons's metrics, the performance of multi-level networked RAID based on cluster is almost same to that of improved 2D-parity. Compared with other structure, it's lower cost and easier to realize.

Keywords RAID, parity, MTTDL

1 Introduction

The users of computers are currently enjoying the unprecedented growth in the past years. In fact, the performance of chips has improved following Moore's Law[1] so far:

$$Transistors / Chip = 2^{year-1964} \quad (1)$$

To maintain the balance of costs in computer systems, Secondary storage must match the advances in other parts of system. A key measure of magnetic disk technology is the growth in the maximum number of bits that can be stored per square inch, or the bits per inch in a track times the number of tracks per inch, called MAD. Its growth followed the "First Law in Disk Density"[2] in the past years:

$$Bits / inch^2 = 10^{(year-1971)/10} \quad (2)$$

The performance of single disk is dominated by the seeking and the rotation delays time. Constrained by the technology, the performance of the seeking and the rotation delays improved slowly (almost 7% per year). It constrained the growth of hard disk performance. With the development of computer technology, the applications that require heavy I/Os become the main applications in the field. The I/O sub-systems become one of the most important parts of the system. So the I/O system becomes one of the most important bottlenecks of computer system.

Constrained by the technology, the performance of seeking and rotation delays improved slowly (almost 7% per year). It constrained the improvement of hard disk performance. So the I/O system became one of the most important bottlenecks in computer system.

RAID (*Redundant Arrays of Inexpensive Disks*) [3] is a solution to the challenge. The data was interleaved on a lot of small low-cost disks by stripe. Benefited from the parallel I/O of the disks, RAID gets large capacity and high performance. With the number of disks increasing, the possibility of failure also increases. Furthermore, a failure will affect multiple files. The two factors impair data availability of system. So RAID adds redundant information to insure data availability. In fact the mass storage systems are based on the RAID now.

[1] This work is supported by the National "863" High-Tech Program of China (No. 863-306-ZD01-02-6)

However, RAID has its drawbacks in terms of cost/performance ratio, availability and scalability. Due to custom hardware, the cost per megabyte of RAID increases with system capacity. Disk arrays need to be connected to a host server, which becomes a bottleneck for both performance and availability. The scalability is also limited by the centralized structure of RAID also. In recent years, with the technical development, the disk density as well as data rates increased quickly. The cost per megabyte of disk fell also, which makes it possible to build low-cost mass storage systems.

All these factors lead to the development of Storage Networks. The researches on the field include SAN (*Storage Area Network*), NAS [4] (*Network Attachment Storage*) and multi-nodes networked storage system. A node includes CPU, memory, bus and disk etc. Any failure of these will leads to the failure of the whole node. So the MTTF (*Mean Time To Failure*) of a node is lower than the MTTF of a disk. Because the RAID only tolerates single disk failure, and it doesn't fit to the networked storage system. It needs some new data layouts to tolerate multiple failures.

2 The structure of multi-level networked RAID

Now the researches on data layout to tolerate multiple failures include grouped RAID, 2D-parity, 3D-parity [5] and Reed-Solomon code etc. The multi-level networked RAID integrates the merit of grouped RAID and 2D-Parity. It's a new method based on Cluster using networked software RAID.

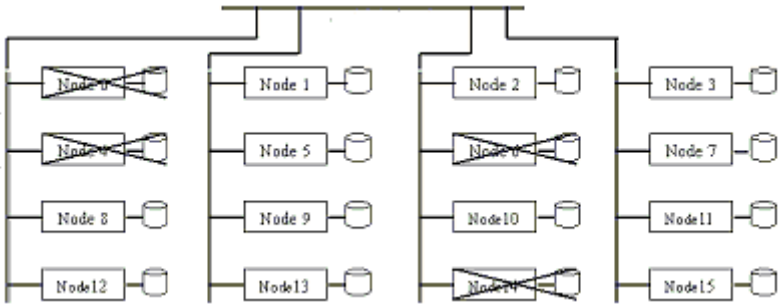

Figure 1 the two-level networked RAID

Figure 1 shows a two-level networked RAID system including 16 nodes. All of the nodes are divided into 4 groups, and each group has four nodes. The disks are attached to the nodes. In default, each node has one disk. The nodes of the same group are connected by a sub-network, and all of the groups are connected by another sub-network. The disks attached to nodes in the same group are organized as a large storage sub-system using RAID 5. All of the sub-systems are organized as a mass storage system, which can tolerate a group of nodes failure.

3 The structure analysis

3.1 Metrics for RAID

There are many metrics that can be used to assess the suitability of a coding scheme in a large disk array. Gibson put forward four metrics in his paper [5]:

MTTDL (*mean time to data loss*)

The MTTDL is a primary metric for a coding scheme. In the manufacture, the MTTDL is estimated by Monte Carlo simulation.

Check Disk Overhead

The *Check Disk Overhead* for a coding scheme is the ratio of the number of parity space to that of data space. It implies the redundancy of the storage system.

Update penalty

The *update penalty* of a coding scheme is the number of parity units whose contents must be changed when the contents of a given data unit is changed. The advantage of large disk arrays lies in the concurrent processing of many random secondary storage accesses. If a code requires N>1 disks to be involved in every write, then the available parallelism is reduced by up to a factor of N. Because parallelism is the reason we want to use disk arrays, the number of disks required to effect a small data update must be minimized.

Group size

The set of disks that must be accessed during the reconstruction of a single failed disk form a group. The *group size* is an important metric because the duration of reconstruction is likely to scale linearly with the number of disks to be read. Additionally, in very large arrays, individual disk failure will be so frequent that highly available systems must continue operation during repair and reconstruction. Until reconstruction is complete, the group size indicates the number of disks that must be accessed to read or write an unreconstructed block on a failed disk. Moreover, the group size also indicates the number of operational disks for which user access performance is degraded by a reconstruction.

Based on the four metrics, the difference among multi-level networked RAID (two-level networked RAID in the paper), RAID 5, grouped RAID 5, 2D-parity and improved 2D-parity is listed. Figure 1, 2, 3, 4, 5 show these cases.

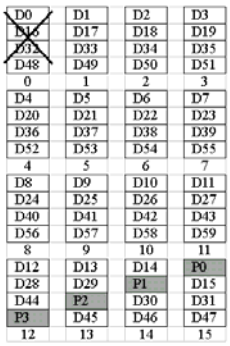

Figure 2 RAID 5

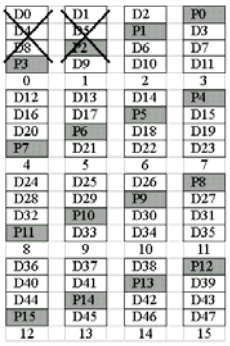

Figure 3 Grouped RAID5

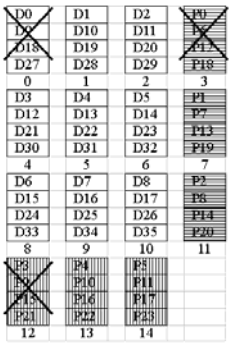

Figure 4 2D-parity

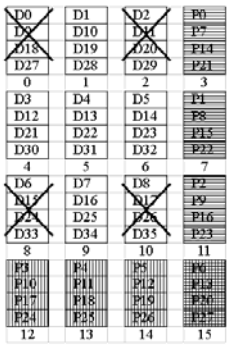

Figure 5 Improved 2D-parity

3.2 The analysis

Fault-tolerant

To describe MTTDL visually, we present *fault-tolerant*. The *fault-tolerant* is the worst case leading to the system crashed.

The RAID 5 can tolerate all single failure. As long as any two disks failed at the same time, the system crashed.

The Grouped RAID 5 can tolerate all single failure and some multiple failure cases. The worse case is: if two disks in the same group failed, the system crashed, just like disk 0 and 1 failed that showed in figure 3.

The 2D-parity can tolerate all double failures and some multiple failure cases. The worst case is: if the parity disks in the same row and column of the failed disk failed simultaneously, the system crashed. In the data layout map, the failed disk and two parity disks form a right-angled triangle as showed in figure 4. P0, P1 and P2 are horizontal parity disks. P3, P4 and P5 are vertical

parity disks.

The improved 2D-parity can tolerant all triple failures and some multiple failure cases. The worst case is: if the quadruple failed disks form a rectangle in the data layout map, the system crashed. As showed in figure 5, the disk 0, 2, 8 and 10 failed simultaneously. In figure 5, P0, P1 and P2 are horizontal parity disks. P3, P4 and P5 are vertical parity disks. P6 is check disk of parity disks.

It's easy to understand that a multi-level networked RAID system has longer MTTDL if it has more levels. In this paper, the two-level networked RAID is compared with other schemes. The worst case of the two-level networked RAID is: in two groups, if there are two nodes failed in the same time, the system crashed. In the data layout map, the quadruple failed disks form a trapezoid. As showed in figure 1, node 0, 4,6 and 14 failed simultaneously.

In storage networks, the network equipments are the single failure point of the system. But it didn't take into account in other schemes. The two-level Networked RAID can tolerate single sub-network failure. So it ensures that the system is available in disaster.

MTTDL

Using probability theory we can calculate the MTTDL. In our model, N is the number of data units in a stripe. G is the number of groups. We assume disk failures are independent and exponential in our models. And MTTR (*mean time to repair*) is exponential also.

$$P_{\{at\ least\ one\ of\ the\ remaining\ disks\ failing\ in\ MTTR\}} = 1 - e^{-MTTR*N\ /\ MTTF} \qquad (3)$$

In all practical cases: MTTR<<MTTF/N ,

and since (1- e^{-x}) is approximately X for 0<X<<1, so

$$P = \frac{MTTR_{disk} N}{MTTF_{disk}} \qquad (4)$$

then $$MTTDL_{RAID5} = \frac{E_{\{Time\ \ between\ \ Failures\}}}{P} = \frac{MTTF_{disk}}{(N+1)P} = \frac{MTTF_{disk}^2}{N(N+1)MTTR_{disk}} \qquad (5)$$

With the same method, we have:

$$MTTDL_{Grouped\ \ RAID5} = \frac{MTTF_{disk}^2}{GN(N+1)MTTR_{disk}} \qquad (6)$$

$$MTTDL_{2D-Parity} = \frac{MTTF_{disk}^3}{GN(N-1)^2 MTTR_{disk}^2} \qquad (7)$$

$$MTTDL_{improved\ \ 2D-parity} = \frac{4MTTF_{disk}^4}{GN(N-1)^2 MTTR_{disk}^3} \qquad (8)$$

$$MTTDL_{Two-level\ \ networked\ \ RAID} = \frac{8MTTF_{node}^4}{G(G-1)N^2(N-1)^2 MTTR_{node}^2} \qquad (9)$$

If we assume that the MTTF and MTTR of disks are equal to that of nodes, we will get

$$MTTDL_{Two-level\ networked\ RAID} = \frac{8MTTF}{(G-1)N} MTTDL_{2D-Parity} = \frac{2}{(G-1)NMTTR} MTTDL_{\text{Im}proved\ 2D-Parity}$$

In practical cases, MTTF is much longer than 50,000 hours and MTTR is shorter than 3 hours. So MTTDL of the two-level networked RAID is much longer than that of 2D-parity. And it is a little shorter than that of improved 2D-parity.

Check disk overhead, Update Penalty and group size

Coding Scheme	Check Disk Overhead	Update Penalty	Group Size
RAID 5	1/15	1	16
Grouped RAID 5	4/12	1	4
2D-Parity	6/9 (total 15 disks)	2	4
Improved 2D parity	7/9	3	4
Two-level Networked RAID	7/9	3	4

Table 1 Comparing Codes for an array of 16 disks (or nodes)

Table 1 shows the difference among the five schemes used in a system including 16 units. It shows that the penalty of two-level networked RAID is equal to that of improved 2D-parity.

4 Experimental results

The experiments are based on the prototype system using two-level networked RAID. It consists of 10 nodes. Each node has four 36GB SCSI disks (IBM DDYS-T36950M). The nodes connect to a Fast Ethernet switch (Cisco Catalyst 3524XL). In the prototype, all of the disks are organized to make a large virtual disk whose capacity is 0.97TB (two-level RAID 5). To compare the performance in different situations, we select three objects in the experiments.

The *Networked RAID* is the prototype that we built. The users can use it through a control server. (In the experiments, we used an IBM NF 7100 as the control server.)

The *SCSI Disk* means a SCSI disk (IBM DDYS-T36950M) attached to the control server.

The *FC-RAID* is a Fiber-Channel hardware RAID attached to the control server. It includes 12 36GB SCSI disks, which are built by RAID 5. The available capacity is 396GB.

The experiments were completed by NetBench 7.0. NetBench is a benchmark program that measures the performance of file servers. It is released by ZD-NET. The server of NetBench runs on a file server, the clients run on some PCs connected to the file server. The clients execute the NetBench tests and send file I/O requests to the server. The NetBench server handles requests from clients and calculates the scores of file server. In the experiments, *Networked RAID*, *SCSI Disk*, *FC-RAID* will be the shared file storage device respectively. By comparing the scores, we can find the difference among *Networked RAID*, *SCSI Disk* and *FC-RAID* to network users.

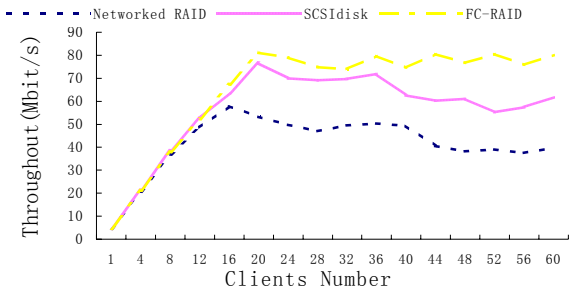

Figure 6 The experimental results

As shown in figure 6, if the number of clients less than 20, then the performance of these systems can't be distinguished. With the clients increasing, it's clear that the performance of *FC-RAID* is the best. Although the performance of Networked RAID is worse than others, it's acceptable to most users.

5 Conclusion

The motivation of this research is to build a high performance and low-cost mass storage system using a cluster of workstations. Based on the metrics of Gibson, it's clear that multi-level networked RAID is a high reliable and low cost scheme. And it doesn't need to design new map functions. It's also easy to realize. It provides scalable, reliable storage system for applications that require heavy I/Os and high transfer rate.

Acknowledgements

We would like to acknowledge the people whose comments about drafts of this paper greatly contributed: Zhou Jie, Shao Xiu-Li, Xiong Wei, Wu Ying and Cui Bao-Jiang. The work described here was supported in part bye the "863" program under grant no. 863-306-ZD01-02-6, as well as support from the TEDA school of Nankai university.

Reference

1. G.E. Moore, Progress in Digital Integrated Electronics, Proc IEEE Digital Integrated Electronic Device Meeting, 1975, pp11
2. P.D. Frank, Advances in Head Technology, presentation at Challenges in Disk Technology Sgort Course, Institute for Information Storage Technology, Santa Clara University, Santa Clara, California, December 15-17, 1987
3. Patterson D A, Gibson G, Katz R H. A case for redundant arrays of inexpensive disks (RAID). In: Proc of 1988 ACM SIGMOD Int'l Conf on Management of Data. New York: ACM Press, 1988,109～116
4. Gibson G A, Nagle D F, NASD scalable storage systems, In: USENIX99, Extreme Linux Workshop, Monterey CA, 1999
5. G. Gibson, L. Hellerstein, R. Karp, R. Katz and D. Patterson, Coding Techniques for Handling Failures in Large Disk Arrays, Technical Report UCB/CSD 88/477, Computer Science Division, University of California, (July, 1988.)