韓國 OA學會 論文誌
第4卷 第2號, 1999. 6.

99-4-2-1-2

단일디스크 고장시 RAID 5의 성능개선을 위한 레벨 전환 기법

전 상 훈*, 정 현 식**

Level Conversion Scheme for Improving Performance of RAID 5 on Single Disk Failure

Sang-Hoon Jeon, Hyun-Sik Chung

요 약

주문형 멀티미디어 시스템 등 사용자 요구에 실시간 응답을 하여야하는 저장시스템 구조에 있어, 단일 디스크 고장시 즉각적인 데이타 복구는 상당히 중요하다. 본 논문에서는 기존의 RAID5구조에 있어 단일디스크 고장시 새로운 디스크가 교체되기 전까지 유발되는 급격한 성능저하를 RAID 레벨 전환 기법을 이용하여 개선하고자 한다. 이 기법은 기존의 연구에서 제시된 예비 디스크 기법에 비해 추가의 디스크를 필요로 하지 않는 저가형 시스템에 적합하다. 제안된 기법과 기존의 RAID5구조는 시뮬레이션을 통하여 여러 모드에서 성능이 측정되었다. 측정 결과 제안된 기법은 실패모드에서는 20%, 재구성모드에서는 80%이상의 성능 개선을 보인다.

Abstract

It is very important to recover data immediately at a single disk failure for critical applications such as multimedia storage systems, real-time systems and so on. As an efficient solution, this paper proposes that RAID level conversion scheme to improve the performance before a failed disk is replaced with a new disk. By using this scheme, it does not require an additional disk to recover data. Comparing with previous studies, this scheme is appropriate to low cost system that has not additional redundant device. The performance of proposed scheme is evaluated and analyzed with that of RAID level 5 for various requested sizes through the simulation. The results show that the performance of the proposed scheme is improved up to 20 percents compared with that of RAID level 5 at the failure mode and 80 percents at reconfigured mode.

* 경동정보대학 전자정보과 전임강사
** 경도대학 사무자동화과 전임강사
논문심사 : 1999. 4.22. 심사완료 : 1999. 5.19

I. Introduction

The primary function of a disk array(RAID) is to increase data availability, to increase total storage capacity, and to provide performance flexibility by selectively spreading data over multiple spindles. The original classification of RAID levels was published in the SIGMOD paper by Garth Gibson and Randy Katz in 1988[1]. The toxonomy roughly classifies RAID architectures according to the layout of data and parity information on disks. Redundant disk arrays are a single fault tolerant, incorporating a layer of error handling not found in non-redundant disk systems. Recovery from these errors is complex because the system may reach to the large number of erroneous states[2]. RAID level 5 architecture provides reliability using the data protection scheme based on parity and it improve performance using the block interleaving scheme by smaller additional costs[3]. The primary weakness of RAID level 3 is to overutilize and writes the parity disk. RAID level 5 overcomes this problem by distributing the parity blocks across all of the member disks; thus all member disks contain some data and some parity. RAID level 5 spreads the parity by putting the parity block for each stripe unit in successive different locations. Both data and parity are evenly distributed throughout the array. A variety of strategies exist to evenly distribute data units and parity units[4]. The more the number of disks on a system increases, the more one fault increases. RAID is a set of disks with redundancy to protect against data loss. Thus, a disk array should be recovered from a single disk drive crash. But if rapid restoration can not be supported, great degrades of performance could be resulted from doubling the access rate to survive disks until the failed disk is replaced by a good disk. It is important that single disk failures are expected to be relatively frequent in a RAID system[5]. RAID level 1(Disk mirroring) is a traditional approach for improving reliability, but calls for using more than 50% of storage capacity[1]. This is the most expensive option we consider since all disks are duplicated, and every write to a data disk is also a write to a duplicated disk[1].

Spare space scheme in disk arrays provides for reconstructing the failed data during the reconstruction process. By doing this, it keeps up the response time on state of failed disk like origin state of system. Recently, most of RAID applied a spared disk on RAID architecture, but additional redundant device requires of a expensive-costed-system with a large number of disks. In this paper, proposed scheme can result in a significant performance hit when a single disk failure. In this scheme, parity blocks of RAID5 architecture is converted into spare blocks when a single disk fails. Strictly speaking, RAID level 5 are converted to RAID level 0 when a single disk fails. In this case, parity blocks can't function as fault-toleranced redundant blocks, but are used as spare blocks. Therefore this scheme can increase performance in reconfigured-mode operation than RAID level 5 architecture and support the speed of large transaction file storage system on single disk failure.

This study can be considered if only disk subsystem performance is under failure. Section 2 presents the previously proposed several spare disk schemes that related our studies and several strategies for efficient

rebuilding of a failed disk. Section 3 presents an outline of the analysis of our proposed scheme. Section 4 and 5 present simulated system and results. Section 6 provides some concluding remarks.

2. Previous approaches to improve failed system

Hot standby disks(on-line spare disk) are used to recover failure disk immediately by additioning usable area in disk array that they automatically rebuilds the contents of the failed disk on the standby disk from the redundant information on the surviving disk. Most simplest one of these schemes, hot sparing scheme(dedicated sparing) is locked on state of not being used during normal operation until failed disk appeared[1]. In a system with n disks, only n-1 disks in the system are utilized during normal operation. Figure 1 shows each column corresponds to a disk and each row corresponds to the data layout for a track on the disks.

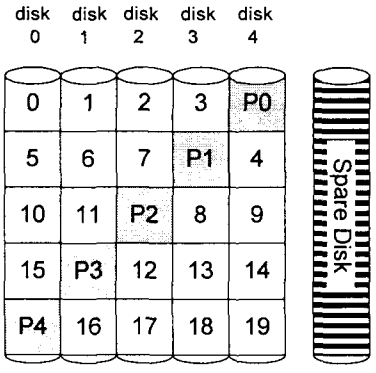


Fig. 1 Parity blocks allocation in hot sparing scheme

Distributed sparing scheme uses the spare space on the disks as a part of workload processing that is distributed on all the disks in the array instead of locating it on a single disk and stored data and parity. Distributed spare space on the disk permits it's recover from a disk failure with no interruption of data availability. Figure 2 shows block location of distributed sparing scheme.

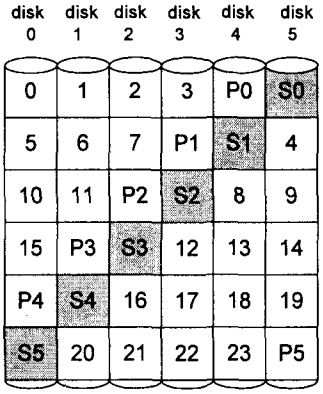


Fig. 2 Spare and Parity blocks allocation in distributed sparing scheme

Compared to the hot sparing scheme, this scheme use all the disks in the array during normal operation, so that it can raise the response time, the total amount of time required to service a request made to a disk array. The response time is composed of three component: queueing time, the time a request spends in a queue waiting to begin execution; positioning time, the time required to position the disk head to useful data; and transfer time, the time required to transfer data to or from the disk.

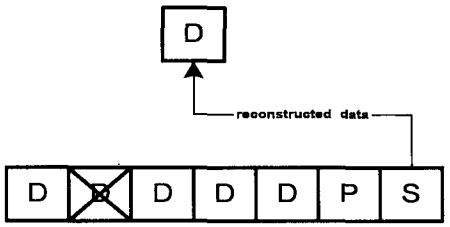
Parity sparing scheme uses the spare space on the disks as a part of secondary parity disk and thus it can reduce the parity group length that means the number of disks in a parity group. When one disk failure is detec-

ted, the parities of two groups are combined to get a single larger parity group with a larger parity group length. This scheme can reduce the parity group length by making effective use of the spare space during the normal operation. Small size parity group length is more efficient to constructing parity disk on normal operation and has better performance in transaction processing applications. Compared to the RAID level 5 disk array, if one of the disks fails, all the surviving disks in the level 5 disk array participate to reconstruct the data on the failed disk, hence these surviving disks observe a load increase of 100% during a failure mode. Figure 3 shows block location of parity sparing scheme.

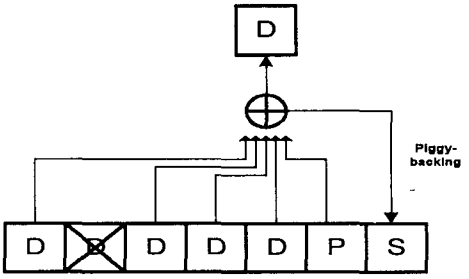
disk 0	disk 1	disk 2	disk 3	disk 4	disk 5
0	1	2	3	P_A	P_B
5	6	7	P_A	P_B	4
10	11	P_A	P_B	8	9
15	P_A	P_B	12	13	14
P_A	P_B	16	17	18	19
P_B	20	21	22	23	P_A

Fig. 3 Parity blocks allocation in Parity sparing scheme

Several disk rebuild strategies for disk arrays are discussed by Muntz and Lui in 1990[5]. In their study, they propose three disk rebuild strategies termed as baseline copy procedure, simply sequentially reads blocks from the failed disk and writes them to the standby disk; rebuild with redirection of reads, a part of read requests to already reconstructed data are supported by reading the data from the spare disk rather than reconstructing the data from the surviving disks again; piggy-backing rebuild, captured a block of the failed disk that is reconstructed due to a read request that was issued as part of the normal workload. Figure 4 shows rebuild with redirection of reads and piggy-backing rebuild strategies.



(a) Rebuild with Redirection of Reads



(b) Piggy-backing Rebuild on Normal Workloads

Fig. 4 Strategies of Disk Rebuild

As a result of these strategies, it can reduce the load on the surviving disks in an disk array with failed disk. Since single disk failures are expected to be relatively frequent in a disk array, all of the these sparing scheme has to required additional redundant device. Therefore expensive cost be supported to traditional disk array systems. This is greatly drawback in inexpensive disks array systems. So, we proposed the cost-effective architecture that can result in a significant performance hit when a single disk failure. We will illustrates the proposed architecture in the next section.

3. Level Conversion Scheme to the cost effective on failed system.

As mentioned in the previous section, single disk failures occurs relatively frequent in a disk array which decrease the response time until the failed disk is replaced by a good disk. To improve this problem, sparing disk schemes are used to process reconstruction immediately from failure disk by additioning usable area in disk array. Therefore, sparing disk schemes are very effective on single disk failures but these schemes require additional disk spaces to maintain each array size. So, we proposed cost effective architecture to bring a significant performance hit when a single disk failure occurs based on the RAID level 5 architecture. In this paper, our scheme's response time is evaluated on various transaction file size with a rebuild strategies. Menon and Matterson categorized a spare disk operation mode: normal mode, the mode is a term during which all the disks in the system are not failed and that is essentially longest mode than any others; failure mode, the mode is a term during which a disk has failed and no reconstruction process initiated and all of the access to the failed disk are supported by the redundant information on the surviving disk that decreases the response time on disk array system; reconstruction mode, the mode is a term during which all the surviving disks in the disk array participate to reconstruct the data on the failed disk hence these surviving disks observe a load increase of 100% during a failure mode; reconfigured mode, the mode is a term after the reconstruction process finished reconstructing the data on the failed disk, but before a new spare is brought into the system to replace the failed disk that is very to be like a RAID level 5; restoration mode, the mode is a term after replace the failed disk to return normal mode, a data and parity in disks is reallocated to restore in a new spare disk[6]. In our scheme, parity space of RAID5 architecture is converted into spare space when a single disk fails. So that, this scheme can increase more reconfigured-mode performance than RAID5 architecture and support a response time of a large transaction storage system on a single disk failure that is illustrated in Figure 5.

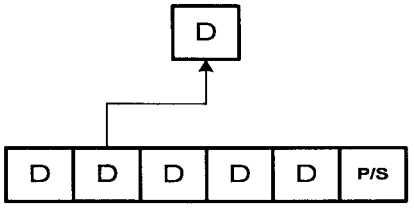
3.1 Normal-mode approach

During normal mode operation, our scheme operate to be alike a RAID level 5. RAID can be Software, Hardware or a combination of both. Hardware RAID offers Parity-based protection. For parity-based protection, exclusive-OR operations are needed. Figure 6 shows a read and write request operation on one parity group in normal mode.

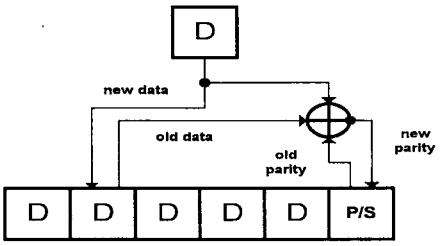
disk 0	disk 1	disk 2	disk 3	disk 4	disk 5
0	1	2	3	4	P/S
6	7	8	9	P/S	5
12	13	14	P/S	10	11
18	19	P/S	15	16	17
24	P/S	20	21	22	23
P/S	25	26	27	28	29

Fig. 5 Spare and Parity blocks allocation in Level Conversion scheme

In Figure 6, D corresponding to data, P/S for our conversion block(hybrid block).



(a) Read request

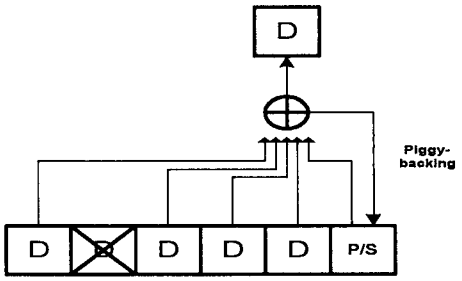


(b) Write request

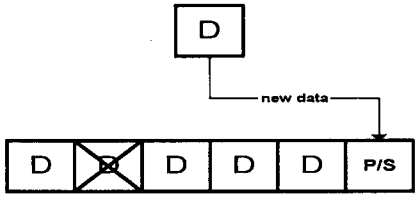
Fig. 6 Read and Write request on normal-mode operation

3.2 Failure-mode approach

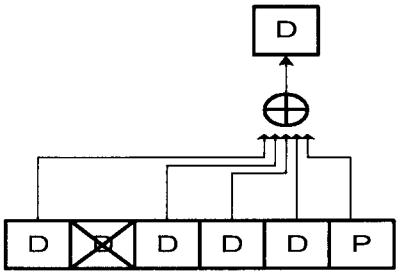
When a read request on failed block is supported, one access rate to hybrid block is added by introducing rebuild strategies for disk arrays, the piggy-backing rebuild. But this access rate can help to reduce the load on the surviving disks in an array during the reconstruction mode. When a write operation is requested on failed block, only one access time is needed on a hybrid block. In RAID level 5 cases, first one access rate is needed on surviving disks to obtain a old data and then second access rate is needed on a parity block to record the new parity data. Figure 7 illustrates variety operation in failure-mode.



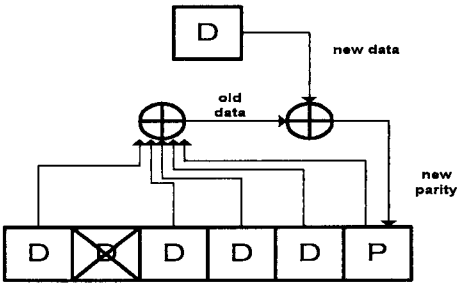
(a) Read request with piggy-backing



(b) Write request in Level Conversion Scheme in Level Conversion Scheme



(c) Read request in RAID level 5

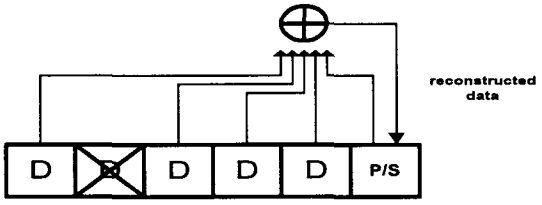


(d) Write request in RAID level 5

Fig. 7 Variety operations in failure-mode

3.3 Reconstruction-mode approach

In RAID level 5 cases, this mode is useless, because there are no spare space to be reconstructed. But in our scheme, parity block in RAID level 5 can be converted to spare space as distributed sparing scheme on single disk failure. Figure 8(a) shows reconstructing on a data block in parity group when single disk fails. This figure to be alike in hot sparing and distributed sparing scheme during the operating. And Figure 8(b) shows reconstructing on a parity block in parity group when single disk fails. In this case, any operations are not occurred, while distributed sparing scheme reconstructs a parity block in a parity group on a spare space.



(a) Reconstruction operation of parity group on failed disk with data block

(b) Reconstruction operation of parity group on failed disk with parity block

Fig. 8 Reconstruction-mode operation in Level Conversion Scheme

3.4 Reconfigured-mode approach

It is the term between after the reconstruction process finished reconstructing the data on the failed disk and before a new spare is brought into the system to replace the failed disk. Like as Figure 7(c), RAID level 5 require multiple reads to the surviving disks in the same array each time. In the worst case, this can double the access rate to the surviving disks[5]. But, our level conversion scheme has a good response time on a workload in this case. Figure 9 shows read request on failed block in reconfigured-mode operation.

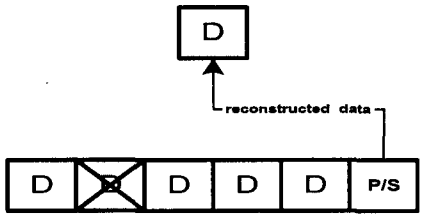
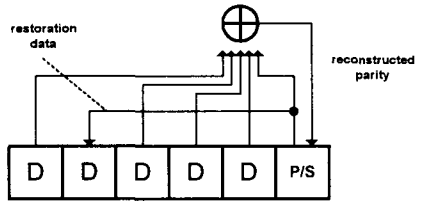


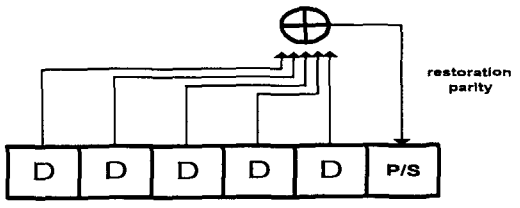
Fig. 9 Read request on failed block in reconfigured-mode operation

3.5 Restoration-mode approach

After replacing the failed disk to return normal mode, a data and parity in disks is reallocated to restore in a new good disk. Figure 10(a) shows restoration process in case of a failed data block in a parity group. In this case, first one access rate is required on surviving disks to reconstruct a parity block and second access rate is required to restored a data on a new good disk to record a reconstructed parity on a hybrid block.



(a) Restoration operation of parity group on failed disk with data block



(b) Restoration operation of parity group on failed disk with parity block

Fig. 10 Restoration-mode operation in Level Conversion Scheme

Figure 10(b) shows restoration process in case of a failed parity block in a parity group that seems like a restoration process in RAID level 5.

4. Analytical simulation modeling

In this section we present a analytical model to simulate our scheme. Our scheme is implemented that used the discrete event simulation library(smpl) based on C language[7]. Table 1 tabulates the parameters of the simulated disk. Disk parameter is based on the IBM 0661 3.5" SCSI disk drive. As input/output type, we treated larger transaction file for super-computing workloads than small bustle transaction file, that because most of the RAID treats these type file. We evaluated our level conversion scheme in file processing time to treat one file of fixed size. Improving the performance in a disk array systems that means to reduce the response time or increase the throughput[8]. Comparing the RAID level 5, we measured the performance rate on single disk failure. Performance rate that can be obtained by using Amdahl's Law.

Speedup = Processing Time(old) / Processing Time(new)

When input/output are treated, that's event are divided into 8 ways[7]. These are organized ① CPU request/release, ② SCSI controller request/release, ③ SCSI Bus request/release, ④ Disk Seek, ⑤ Rotate waiting, ⑥ SCSI Bus request/release, ⑦ Data Transfer, ⑧ SCSI Bus release. Figure 11 shows events diagram that each event are processed sequentially.

Table 1 Disk Parameters

cylinders per disk	949
tracks per cylinder	14
sectors per track	48
bytes per sector	512
disk capacity	311MB
revolution time	13.9ms
single cylinder seek time	2.0ms
average seek time	12.5ms
max stroke seek time	25.0ms
max sustained transfer rate	1.7MB/s

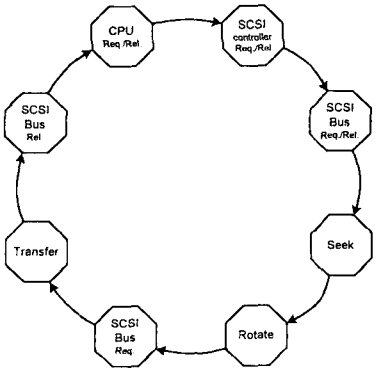


Fig. 11 Events Diagram

File processing time is like to execute once cycle processed of events diagram sequentially. Simulated disk array is constructed with six array width and each parity blocks in a disk array is allocated according to left-symmetric parity distribution method[8]. The left-symmetric placement is derived by left rotations of entire parity stripes from the RAID level 4 placement. E. K. Lee's paper shows that left-symmetric is the best RAID level 5 parity placement. Seek times are calculated with the following equation that using the nonlinear model[4]:

$$seekTime(x) = \begin{cases} 0 & \text{if } x = 0 \\ a\sqrt{x-1} + b(x-1) + c & \text{if } x > 0 \end{cases}$$

When x is the seek distance in cylinders and a, b and c are chosen to satisfy the single-cylinder-seek-time, average-seek-time and max-stroke-seek-time constraints. If cylinders-per-disk is greater than approximately 200, a, b and c can be approximated using the following formulas:

$$a = (-10\min Seek + 15 avgSeek - 5\max Seek)/(3\sqrt{vmCyl})$$

$$b = (7\min Seek - 15 avgSeek + 8\max Seek)/(3vmCyl)$$

$$c = \min Seek \text{(single cylinder seek time)}$$

For using the above disk parameter, a = 0.4623, b = 0.0092 and c = 2.

5. Performance evaluation

This section analyzes the performance of the proposed level conversion scheme approach using the analytical simulation model. We assumed a seek cost function is nonlinear and disks in a parity groups are synchronized. Normal requests are assumed to be read requests with 70% probability and write requests with 30% probability. Requests are assumed to arrive with an exponential distribution.

In normal operation, response times are equal to RAID level 5. That's reason, our scheme uses the same size as parity group of RAID level 5 architecture. File processing times, during failure operation, of the two architectures considered in this paper are shown in Figure 12.

Figure 12 illustrates file processing times during failure-mode. Since the reliability of the disk array is quite dependent on the reconstruction time[9], our study employ rebuild strategies with redirection of reads and piggy-backing. In failure-mode operation, our scheme approach has better file processing time than RAID level 5 architecture. The processing time on performance is more pronounced at higher loads.

In reconstruction-mode operation, RAID level 5 is useless because there are no spare space to reconstructed. But in our scheme, parity block in RAID level 5 is converted to spare space as distributed sparing scheme on single disk failure. After the reconstruction process finished reconstructing the data on the failed disk, a level conversion scheme has better performance than RAID level 5 architecture that shown in Figure 13. Because of request redirection, the normal requests to already reconstructed data on a hybrid block get serviced quicker than RAID level 5 architecture.

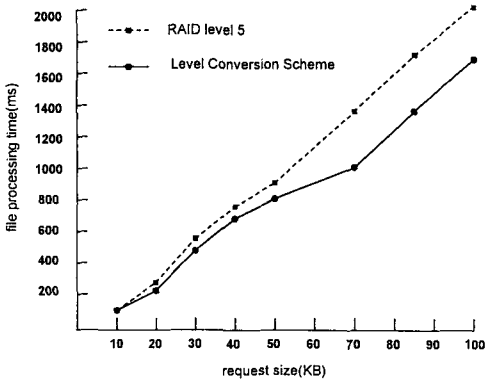


Fig. 12 File processing times during failure-mode

Larger transaction files in simulation are used in reconstructing data from the failed disk takes more number of operations. Hence, converted spare disk on hybrid block is very useful on a single disk failure. Since the reliability of the disk array is quite dependent on the reconstruction time[10], if the period of replacing the failed disk to return normal mode is not shorted, this problem has a worse performance in a disk array system.

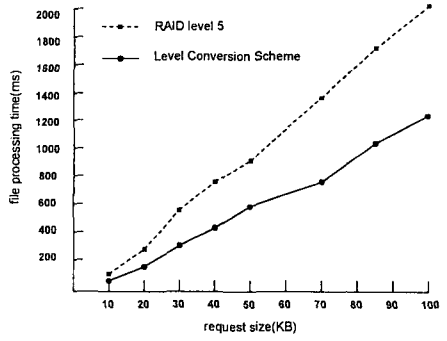


Fig. 13 File processing time during reconfigured-mode.

6. Conclusions

RAID level 5 disk arrays interleave data across multiple disks in blocks called striping units. To protect against single-disk failures, RAID level 5 adds a parity block for each row of data blocks. These parity blocks are distributed over all disks to prevent any single disk from becoming a bottleneck[11]. This fault tolerance scheme is very effective on cost and capacity without any additional spare space, but in case of continuous read requests on a failed disk, many times access rate has to supported on surviving disks to satisfied these requests. This can double the access rate to the surviving disks for a workload of all reads. Therefore increased response times are serious problem on single disk failure in RAID level 5[10]. Performance of proposed scheme in this paper can increase up to 20 percents at failure mode and 80 percents at reconfigured-mode than RAID5 architecture and support the speed of a large transaction file storage system on a single disk failure. As read requests are increases, performance are more improved up in most of multimedia storage systems such as VOD (Video On Demand). Using of RAID controller with non-volatile cache, restoration times of 3GB disks are less than 20 minutes[6]. So, we need not consider its overhead.

To the conclusion, proposed scheme has spare space that provided for reconstructing the failed data during the reconstruction process without additional redundant device being required. We evaluated the performance of new organizations in various transaction file size of operation, with a rebuild strategies.

References

[1] D. A. Patterson, G. A. Gibson, and R. H. Katz, "A Case for Redundant Arrays of Inexpensive Disks(RAID)," International Conference on Management of Data (SIGMOD), pp. 109-116, June. 1988.

[2] W. V. Courtright II, G. A. Gibson, "Backward Error Recovery in Redundant Disk Arrays," Technical Report REF42170, Carnegie Mellon University, 1994.

[3] D. Stodolsky, G. A. Gibson, and M. Holland, "Parity Logging Overcoming the Small write Problem in Redundant Disk Arrays," Proceeding of the 20th Annual International Symposium on Computer Architecture, pp. 190-199, May. 1993.

[4] E. K. Lee, "Performance Modeling and Analysis of Disk Arrays," Ph.D Thesis, Carnegie Mellon University, 1994.

[5] R. R. Muntz and J. Lui, "Performance Analysis of Disk Arrays Under Failure," Proceedings of 16th VLDB Conference, pp. 162-173. 1990.

[6] J. Menon and R. Mattson. Comparison of sparing alternatives for disk arrays. Proceeding of International Symposium on Computer Architecture, May. 1992.

[7] M. H .MacDougall, "Simulating Computer Systems," MIT Press, 1987.

[8] E. K. Lee, R. H. Katz, "Performance Consequences of Parity Placement in Disk Arrays." International Conference on Management of Data(SIGMOD), pp. 190-199. 1991.

[9] J. Chandy and A. L. Narasimha Reddy, "Failure Evaluation of Disk Array Organization," Proceedings of the International Conference on Distributed Computing Systems, IEEE Computer Society, Washington D.C., 1993.

[10] E. K. Lee, "Software and Performance Issues in the Implementation of a RAID Prototype," Technical Report UCB/CSD 90/573, University of California at Berkeley, May. 1990.

[11] P. M. Chen, E. K. Lee, "Striping in a RAID Level 5 Disk Array," Technical Report University of Michigan, 1993.

저자소개

전상훈

1992년 2월 영남대학교 전산공학과 졸업(공학사)

1994년 2월 영남대학교 대학원 전산공학과 졸업(공학석사)

1998년 8월 영남대학교 대학원 전산공학과 박사수료

1999년 3월~현재 경동정보대학 전자정보과 전임강사

관심분야 : 멀티미디어 시스템, 정보통신, 컴퓨터 구조

정현식

1987년 2월 경일대학교 전산공학과 졸업(공학사)

1990년 2월 영남대학교 대학원 전자공학과 졸업(공학석사)

1997년 2월 영남대학교 대학원 전산공학과 박사수료

1999년 3월~현재 경도대학 사무자동화과 전임강사

관심분야 : 이동통신 데이터베이스, 분산처리 시스템,컴퓨터 구조