A Fault-tolerant Distributed

Disk Cache System for Redundant Disk Arrays

School of Information Science
Japan Advanced Institute of Science and Technology
Makoto Kojima

keywords RAID, reliability, disk cache,
hit ratio, disk access , performance

Abstract

1 Introduction

Recently, computer systems tend to be enlarged more and more to meet demands of performance on information processing. Performance of CPU and semiconductor memory continue to be enhanced by improving processor architecture and technology. However, to obtain high performance of total computer systems, it is also required to improve the performance of I/O subsystems.

RAID (Redundant Arrays of Inexpensive Disks) has been proposed as a disk array system to satisfy those demands for secondary storage systems. RAID provides high availability by using parity encoding of data to survive disk failures. However, one of the serious problems RAID has is too frequent disk accesses: RAID level-5 array needs four disk accesses in order to update a data block – two for reading old data and parity, and the other two for writing new data and parity[1][2].

A scheme perviously used to improve the update performance of such arrays is making use of a large capacity disk cache for RAID. But the scheme does nothing special in order to suppress any cache faults.

One approach is to mirror the disk-cache with respective power control[2][3]. Now, we consider another approach, called "distributed disk-cache for RAID," which has fault-tolerance for the disk-caches. The distributed disk-cache for RAID is a set of caches and

controllers which can automatically recover data within the system: If one cache in the set fails, it will be recovered by using redundant data maintained by other caches or disks.

2 Overview of distributed disk-cache system

A distributed disk-cache system for RAID is structured by disk drives and the same number of chaches as of the disk drives.

In the distributed disk cache system for RAID, the array-controller has to calculate its parity data at first for the updated data incoming from host. It must be done before stacking it in cache. Thus, the data on cache already has a redundancy based on parity-data. As of this time, the data is provided fault-tolerance. If cache failure occurs after that,this system whould be able to restore it by calculating exclusive OR of other data.

The calculated parity data is put into cache while the updated data is done. The old-data used for calculation are read from each disks or caches. In case of both old-data and old-data being in cache,this update is done quite quickly. If one data is hit and the other miss, the hit cache can send the old-data to the array-controller quickly , hence the array-controller may start forwarding the XOR process without waiting time for disk access.

Caches are conected by line with each other. So they are able to exchange the information of cache data. This information is useful in case of cache failure. It is helpful to reconstruct the lost cache data so quickly.

3 Evaluation of an Experimental System

In this study we prepare the following three systems without disk failures and cache failures ; and evaluate response time , hit ratio and cache model for them respectively: the non-cached RAID 5, with mirrored double disk-cache model and with distributed disk-cache system. These evaluaton devices were composed of some units of transputers. For this evaluation, we used six disk drives, each of which has the capacity of 20MB. And the maximum capacity of the total cache is 1MB.

Our experiments indicate the following features. When we say "hit for the distributed disk-cache system" in the following, we mean that the updated data and parity data are hit in caches at the same time in the system.

- **differences of hit ratio**.

  Firstly, we discuss writing operation for systems. The difference of hit ratio between the distributed disk-cache system and the double disk-cache system is very little when the fully associative method is adapted. And hit ratio for either system is increasing

in propotion to the total cache capacity of systems. If we apply the direct-map method to these systems, the hit ratio is lower than that when we apply the fully associative method. The hit ratio for either system is from 50% to 75% and it varies widly in propotion to the total cache capacity of systems.

Secondly, we discuss reading operation for systems. In case of the distributed disk-cache system with the fully associative method and the double disk cache system with the fully associative method or the direct map method being compared, either hit ratio is very near to that of the writing operatrion,while the distributed disk-cache system with the direct map method is considered, the hit ratio is near to that of the writing operatrionwhen with the fully associative method.

- **process time**.

  Firstly, we discuss writing operation for systems. In case the distributed disk-cache system is adapted, calculating its parity is done whenever updating the data is needed. So when its cache hit occurs, its processing time is shorter than that of non-cached RAID5 by two third. In case the double disk-cache is adapted, it has very short process time when its cache hit occurs.

  But in case its cache miss occurs, the double disk-cache system is slower than the distributed disk-cache system, because the double disk-cache system has to calculate the parity and then destage the data. We compare their search time . The double disk-cache system is slower in searching than the distributed disk-cache system.

  Secondly, we discuss reading operation for systems. When the system conducts reading operation, it need not calculate the parity, so either the distributed disk-cache system or the double disk-cache system takes shorter process time.

- **A running performance test**.

  Next, we have measured the total access time when we keep the system in action. These access have been done at random as writing test and reading test for 25 thousand times respectively.

  Firstly, we discuss the sum of writing operations for systems. The summing up access time by the distributed disk-cache system is lower than that by the non-cached RAID system. As to the summing up access time by the distributed disk-cache system, it was decreasing in inverse proportion to the total cache capacity of the system.

  As to the summing access time by the double disk-cache system with the direct-map method , it is decreasing in propotion to the total cache capacity of systems. But as to the summing access time by the double disk-cache system with the fully associative method, it is increasing in propotion to the total cache capacity of systems. Why this cache access is slow is that it is provided as software in our model. But, when we

adopted the direct-map method, as to the difference of summing access time between the distributed disk-cache system and the double disk-cache system, either system being used with the method, such difference is not increasing in propotion to the total cache capacity of either system.

Secondly, we discuss the summing up reading operations for systems. As to the difference of the summing access time between non-cache RAID and other systems, the value is so little when it compares with writing operations. As to the value of the summing access time , it was so little when it compared with writing operations.

As to the summing up access time by the distributed disk-cache system, if it is used with the direct-map method, the value is decreasing in propotion to the total cache capacity of systems, while if it is used with the fully associative method, the value is increasing.

4 Conclusion

In this research, we have contrived an idea about the distributed disk cache for RAID with a mechanism of fault-tolerance. And we have measured the access speed and the hit ratio when the system has no damege.

The result is as follows: in case of writing the distributed disk cache system, the speed is faster than that of non-cache RAID. In case of comparing it with the double disk-cache, writing speed in the distributed disk-cache system is faster than that of double disk-cache, when the hit ratio is low. Why the distributed disk-cache system is fast is that this system can write cache data to disk without waiting the completion of system XOR operation.

In our experiments, access time in the distributed disk-cache system is not reduced, although the total cache memory is increased. But if we utilize more effective cache search method, the total access time will be more shoter than now.

Performance measurement with failures is one of future researchs.

References

[1] Menon.J and Cortny.J, "The architecture of fault-tolerant cached RAID controller", *IEEE.NEW York,76-8,* 1993.

[2] Peter.M.Chen etal, "RAID:High-Performance,Reliable Secondary Storage", *ACM computing Surveys Vol.26 No2 ,* June,1994.

[3] JAI.MENON, "Performance of RAID5 Disk Arrays with Read and Write Caching", *Distributed and Parallel Database,2,261-293,* 1994.