Rotational Lease: Providing High Availability in a Shared Storage File System*

Byung Chul Tak[1], Yon Dohn Chung[2,**] , Sun Ja Kim[1], and Myoung Ho Kim[3]

[1] Embedded S/W Div., Electronics and Telecommunications Research Institute,
161 Gajeong-dong, Yuseong-gu, Daejeon, Korea
{bctak, sunjakim}@etri.re.kr
[2] Dept. of Computer Engineering, Dongguk University,
26, 3-ga, Pil-dong, Chung-gu, Seoul, Korea
ydchung@dgu.edu
[3] Division of Computer Science, Korea Advanced Institute of Science and Technology,
373-1 Guseong-dong, Yuseong-gu, Daejeon, Korea
mhkim@dbserver.kaist.ac.kr

Abstract. Shared storage file systems consist of multiple storage devices connected by dedicated data-only network and workstations that can directly access the storage devices. In this shared-storage environment, data consistency is maintained by lock servers which use separate control network to transfer the lock information. Furthermore, lease mechanism is applied to cope with control network failures. However, when the control network fails, participating workstations can no longer make progress after the lease term expires. In this paper we address this limitation and present a method that enables network-partitioned workstations to continue file operations even after the control network is down. The proposed method works in a manner that each workstation is rotationally given a predefined lease term periodically. We also show that the proposed mechanism always preserves data consistency.

1 Introduction

Wide-spread use of Internet and increased computing power enabled applications to process more data and file systems have accordingly evolved to utilize network technologies. Recently the development of new medium such as fiber channel has enabled another form of shared storage network file system such as Storage Area Network[1,2,3,4,5]. Shared storage file systems use dedicated network for data traffic to enhance the performance. Storages are transparently shared so that workstations or servers attached to the storage area network use them as if they are local devices. This structure effectively eliminates the possible bottleneck point which was a major drawback of conventional network attached storage. Servers have local cache to store data read from the storages. However, the use of cache inevitably raises cache coherency and data consistency problems. To cope with this problem some of the servers are

* This work was done as a part of Information & Communication Fundamental Technology Research Program, supported by Ministry of Information & Communication in Republic of Korea.
** Corresponding author.

L.T. Yang et al. (Eds.): HPCC 2005, LNCS 3726, pp. 200 – 205, 2005.

are designated as lock managers. Lock managers use control network, which is usually an Ethernet connecting the servers, to transmit lock information.

For network file systems it is crucial that the file system be able to continuously provide reliable services. Among many possible failures in the shared storage environment, our approach deals with the failure of the control network. Conventional approach uses leases to reclaim the locks thereby forcing disconnected servers to stay idle until the problem is physically taken care of, whereas our approach allows servers to continue using the file system.

The remainder of this paper is organized as follows. Section 2 introduces some related works. Section 3 describes the proposed method in detail. Section 4 discusses experiments and analyzes characteristics of the proposed method. Finally in Section 5 we conclude with summary and future directions.

2 Related Work

The shared storage file system requires a mechanism to control access to the shared storage and this is commonly achieved by locking mechanism. But usually lock alone is not enough and the lease[6] mechanism is used together. Lock managers issue per-system lease and if the lease is not renewed within the lease term, lock managers invalidate all the locks issued to the server. StorageTank[2] uses per-system lease for data coherency and opportunistic renewal method in order to minimize the lease renewal cost.

Among many possible failures of the shared storage file system we are mainly concerned with the failure of the control network. Failure of the control network implies that servers are partitioned into several groups. In GPFS[4] only the majority partition is allowed to continue using the storage device. In order to determine the majority partition, GPFS[4] employs a kind of group membership protocol. It also uses a fencing function provided by the SAN switch to block I/O requests from the minority group. CXFS[5] adopts similar group membership protocol. This group membership technique has a drawback of only allowing certain server to survive while others stop indefinitely. And since GPFS[4] and CXFS[5] both uses locking mechanism, they are unable to reclaim the issued locks.

3 The Rotational Lease

3.1 Network Failures in the Shared Storage File System

The shared storage file system uses separate network for data traffic between servers and storage devices to gain performance enhancement. Lock information among servers or file requests from clients are transmitted through the local area network which acts as a control network as in Figure 1 (a). The failure of the control network creates network partitions and partitioned servers can no longer exchange lock information with the lock managers or renew the lease. Although some servers on the same partition as the lock managers may still be able to continue to work, the file system as a whole does not function any more.

Our proposed method tries to overcome this limitation by enabling the file system to continue working after the network partitioning occurs. The proposed method extends the lease concept and regards leases as being revived periodically after it expires. The idea is that participating servers can continue to use the file system if their lease is renewed automatically by predetermined order and time duration. This way of using extended lease concept is in effect making servers to wait for their turn to use the storage in rotation, and in this sense we call it a **Rotational Lease** method.

3.3 Periods and Active Intervals of the Rotational Lease

Automatic and periodic renewal of the lease requires that the renewal order be predetermined and servers be able to correctly find the time interval of their renewed leases. To do so we first need to introduce the concept of active intervals and periods in the Rotational Lease. Figure 1 (b) illustrates the concept.

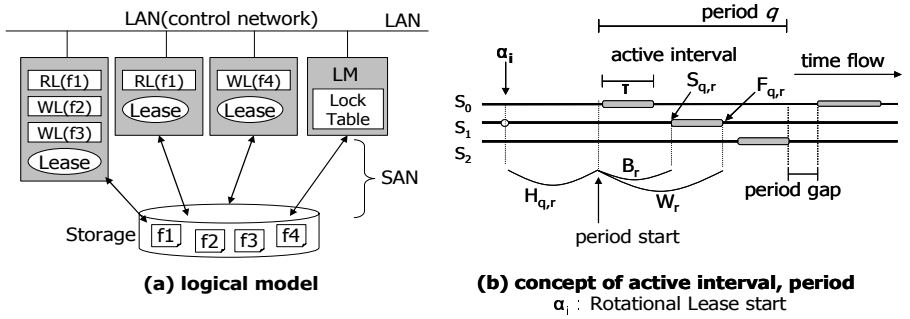


Fig. 1. Logical model of the shared storage file system and the concept of the active interval and period of the Rotational Lease

Determining the active interval involves time computation based on each server's internal clock. But some complication arises from the fact that real clocks deviate from the absolute time and the amount of deviation differs slightly from clock to clock. We define the clock drift rate as the maximum ratio of relative clock speed among the servers' internal clocks per unit time similar to the relative clock rate of StorageTank[2].

$\boldsymbol{\rho}$: Maximum clock drift rate in range of ≥ 1. The time length t at one server falls within the $(t/\rho, t\rho)$ range at another.
$\boldsymbol{\tau}$: Time length of the active interval, or the Rotational Lease term.
$\mathbf{S_i}$: *i*-th server.

The start time of the active interval can be found in two steps: finding the start time of the period and determining the time to wait to enter the active interval from the start of the period. According to Figure 1 (b) the value of interest is the start time of the active interval which is denoted by $S_{q,r}$ and we could establish the following equation.

$$S_{q,r} = \alpha_i + H_{q,r} + B_r \tag{1}$$

The constant α_i of Equation 1 represents the time point where S_i enters the Rotational Lease mode. The active intervals obtained from Equation 1 must possess the following property to ensure the data consistency.

Definition 1. Non-overlapping property : Let $S_{p,r}$ and $S_{q,s}$ be two arbitrary active intervals where *p, q* are the period numbers and *r, s* the active interval numbers and $S_{p,r} < S_{q,s}$ holds. If $F_{p,r} <= S_{q,s}$ is satisfied, then it is said to have the non-overlapping property.

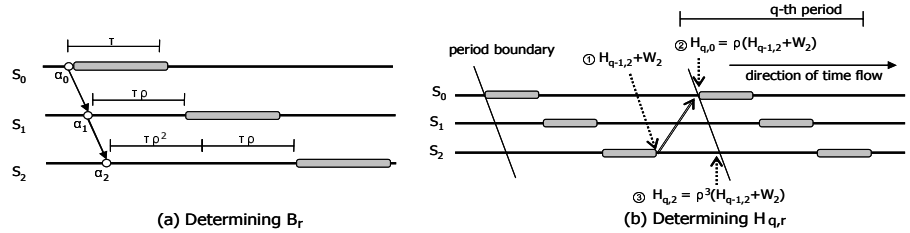


Fig. 2. Determining B_r and $H_{q,r}$

$H_{q,r}$ and B_r on the right hand side of Equation 1 needs to be rewritten in terms of constants, period numbers and active interval numbers. Figure 2 (a) illustrates the idea of transforming B_r and $H_{q,r}$. Generalizing in terms of active interval number *r* gives us equations for and they are used to claim the following theorem.

Theorem 1. The start time of the active interval acquired by the following equation satisfies the non-overlapping property.

$$S_{q,r} = \alpha_i + H_{q,r} + B_r \text{ where } B_r = \tau\sum_{x=1}^{r}\rho^x \text{ and } H_{q,r} = \rho^{r+1}\left(H_{q-1,m-1} + W_{m-1}\right)$$

3.4 Detection of the Failure

For detecting the moment of failure, we employ heartbeat messages that periodically visit every server once and return to the originator. CXFS[5] and GPFS[4] also use heartbeat messages for failure detection. Heartbeat messages are propagated in the order of active interval assignment. The event of network failure is not known to any servers at the very moment. It is not until the next heartbeat message is initiated that servers detect the network failure. Message originator sends out the message and waits for the return. If the message is not returned within appropriate time limit, it concludes that a network problem has occurred. Other servers detect the failure if the heartbeat message does not arrive from the predefined predecessor within the expected time limit.

3.5 Determining the Rotational Lease Parameters

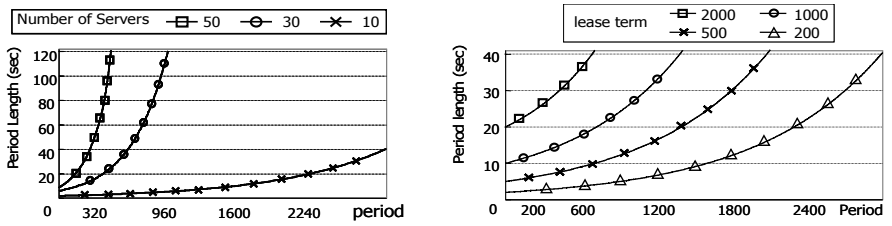
Since smaller number of active intervals means better throughput, it is desirable to minimize the number of active intervals by assigning as many servers as possible to a

single active interval. One way of determining whether two servers have no overlapping region of data set is using the directory structure of the file system. Servers declare their data region by designating a set of directories and it means that the server will only use the data within the subtree. Once this data region is declared, it is transformed into the parent-child relationship graph. In this graph, nodes represent servers and edges represent the parent-child relationship. From this parent-child relation graph we apply the graph coloring algorithm to find the optimal active intervals. The presence of edge means that two nodes have a common data region, and thus they must not be in the same active interval. It is equivalent to assigning a color to a node and assigning different color to adjacent nodes with minimal number of colors. It is known that the graph coloring problem is NP-complete[7]. Many heuristics have been developed, and any of these algorithms can be readily applied to find practical number of active intervals.

4 Simulation and Analysis

Table 1. Parameter used and their value range

Parameter	Actual Value Range	Value Range used
Drift rate	$1+1x10^{-3}$ ~ $1+1x10^{-6}$ sec/sec	$1+1x10^{-3}$ ~ $1+1x10^{-6}$ sec/sec
Number of Servers	10 ~ 100	10, 30, 50
Active interval length	Several msec ~ several sec	200, 500, 1000, 2000 msec



(a) Varying the number of servers (lease term: 200ms, Drift rate:1.0001)

(b) Varying lease term length (Num of servers: 10, Drift rate:1.0001)

Fig. 3. The effect of varying parameters

The behavior of the proposed method is examined using a software simulator. In the simulation, we examined the effect of varying parameters. The parameters used are the number of systems, Rotational Lease term, and clock drift rate. Table 1 shows the actual range of these parameters and the value range used in the experiment.

Figure 3 (a) shows the effect of varying the number of servers, and Figure 3 (b) shows the effect of varying lease terms. We see that larger number of servers and larger lease term both cause the period gap to widen faster, but the number of system affects more than the length of lease term. Note that the period length shows an exponential growth. However according to our simulation it showed that the performance

degradation due to exponential increase of period length is not serious. The period length stays reasonably low long enough that it gives enough time for physically recovering from the network failure.

5 Conclusion

We proposed a method that improves availability of the shared storage file system by enabling it to continue functioning under the failure of the control network. The proposed method, which we call the Rotational Lease, extends the conventional lease concept by automatically renewing the lease in a predetermined order in the event of network failure. Each server with pre-assigned active interval number claims an access right to the storage during the renewed lease term in rotation.

In order to properly apply the Rotational Lease, other conditions must be considered. It was assumed in the paper that participating servers were able to finish up the file operations within the active interval. However, it is possible that servers may fail to do so due to many causes. In future work some measures will be added to protect data consistency in these exceptional cases.

References

1. Steven R. Soltis, Thomas M. Ruwart, Matthew T.O'Keefe, The Global File System, Proceedings of the Fifth NASA Goddard Space Flight Center Conference on Mass Storage Systems and Technologies, Sept 17-19, 1996
2. R. C. Burns, R. M. Rees, and D. D. E. Long, Safe Caching in a Distributed File System for Network Attached Storage, In Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS), IEEE, 2000.
3. Chang-Soo Kim, Gyoung-Bae Kim, Bum-Joo Shin, "Volume Management for SAN environment", In Proceedings of the International Conference on Parallel and Distributed Systems, 2001
4. Frank Schmuck and Roger Haskin. GPFS: A Shared-Disk File System for Large Computing Clusters. Proceedings of the Conference on File and Storage Technologies (FAST'02), pp. 231-244, 2002
5. CXFS: A high-performance, multi-OS SAN file system from SGI. SGI White Paper. URL http://www.sgi.com/products/storage/tech/file_systems.html.
6. C. Gray and D. Cheriton, "Lease: An efficient fault-tolerant mechanism for distributed file cache consistency," Twelfth ACM Symposium on Operating Systems Principles, pp. 202 210, 1989.
7. Michael R. Garey and David S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Comleteness. W.H.Freeman, 1979.