ANALYZING CHARACTERISTICS OF PC CLUSTER CONSOLIDATED WITH IP-SAN USING DATA-INTENSIVE APPLICATIONS

Asuka Hara
Graduate school of Humanities and Science
Ochanomizu University
2-1-1, Otsuka, Bunkyo-ku, Tokyo, Japan
email: asuka@ogl.is.ocha.ac.jp

Kikuko Kamisaka
Natipnal Institute of Information
and Communications Technology
4-2-1, Nukui, Koganei-city, Tokyo, Japan
email: kikuko.kamisaka@nict.go.jp

Saneyasu Yamaguchi
Department of Computer Science
and Communication Engineering
Kogakuin University
1-24-2, Nishishinjuku, Shinjuku-ku, Tokyo, Japan
email: sane@cc.kogakuin.ac.jp

Masato Oguchi
Graduate school of Humanities and Science
Ochanomizu University
2-1-1, Otsuka, Bunkyo-ku, Tokyo, Japan
email: oguchi@computer.org

ABSTRACT
In the information society in recent years, the volume of data requested to process increases explosively. Information required from users should be extracted from it instantly. In order to process huge volumes of data, we have constructed a PC cluster consolidated with IP-SAN that integrates the front-end and the back-end networks into the same IP network. However, the detailed analysis is not yet performed how the communication between nodes of the cluster and the execution of I/O influence the behavior of the system performance, when data-intensive applications are executed on it.

Thus, in this paper, a PC cluster consolidated with IP-SAN is evaluated when the network and target of the cluster are burdened with a heavy load, by accessing a single target with multiple initiators. As an example of data-intensive applications, parallel association rule mining is executed. As the system is monitored and analized, detailed behavior of the PC cluster consolidated with IP-SAN is clarified.

KEY WORDS
IP-SAN, iSCSI, PC cluster, data mining

1 Introduction

With the recent price plummet and performance improvement of commoditized hardware of personal computers (PCs), high-performance data-intensive applications such as large-scale scientific computation, databases, and data mining have been executed on PC clusters. In large-scale PC clusters, high-speed proprietary networks such as Fibre Channel (FC) and InfiniBand have often been used as an access network between a cluster node and storage. However, the advent of IP-based Storage Area Network (IP-SAN), e.g. Internet SCSI (iSCSI) protocol, allows to construct PC clusters only with a network based on commoditized technology.

In this paper, we have proposed a PC cluster consolidated with IP-SAN and evaluated it with data-intensive applications. That is to say, recent PC clusters have a back-end SAN between a node and storage as well as a front-end LAN among nodes. Both networks can be consolidated by using IP-SAN on the PC cluster, which leads to the reduction of construction and operational management costs of a network. We have evaluated such types of clusters especially when their storage is heavily accessed by data-intensive applications.

As a general way of using iSCSI, it is common to access a single target (storage) from multiple initiators (servers). Thus, in this paper, by changing the number of initiators connected to a target, we have observed the behavior of a PC cluster consolidataed with IP-SAN when a heavy load is applied to the network and the target. The system is evaluated with a storage benchmark and parallell data mining executed on it.

2 PC cluster consolidated with IP-SAN

2.1 Using SAN in PC cluster

In recent years, since the quantity of the data processed in an information system has become huge, SAN is introduced to connect storage to such a system, and it has come to be widely used. In the HPC field, SAN is used as a back-end network between a compute node (server) and storage of a PC cluster. SAN unifies the dispersed storage held by each node and realizes efficient practical use of central control for disk resources.

Figure 1 shows an example of a PC cluster connected with SAN. At present, FC-SAN is popular, which is a high-speed dedicated line using Fibre Channel. However, in the case of FC-SAN, it has an obstacle to introduce it into a PC cluster because the cost of FC switch is high.

IP-SAN is called SAN of the next generation built with a commoditized TCP/IP network. Instead of using the conventional SAN built with FC, introduction and employment of IP-SAN as the back-end network reduce the storage cost of a PC cluster. Since 10Gigabit Ethernet will

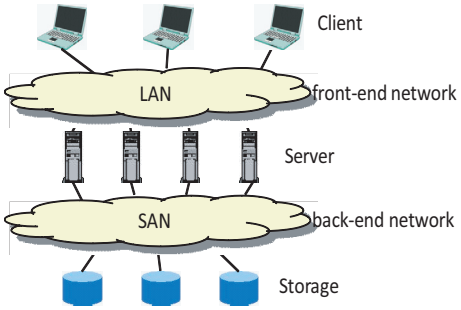

Figure 1. SAN-connected PC cluster

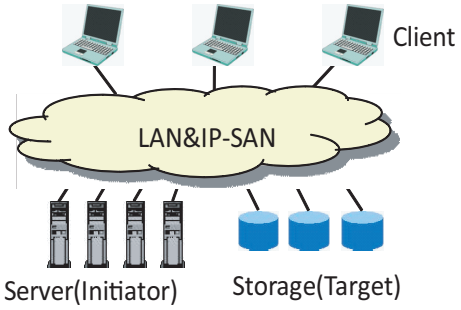

Figure 2. PC cluster consolidated with IP-SAN

be widely used in the near future, PC clusters having IP-SAN at the back-end must come to be popular.

2.2 PC cluster consolidated with IP-SAN and its performance concern

We are evaluating the PC cluster consolidated with IP-SAN which unified the networks of the back-end between a compute node (server) and storage to the front-end between nodes, as shown in Figure 2. iSCSI is the protocol of IP-SAN ratified by IETF in Feburary 2003[1], and data transfer is performed by encapsulating the SCSI command in a TCP/IP packet. Many literatures have discussed about performance of iSCSI protocol[2][3][4]. Figure 3 shows the layered structure of iSCSI.

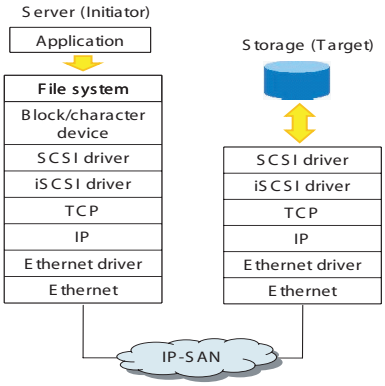

Figure 3. the layered structure of iSCSI

In the case of a PC cluster using SAN, since front-end LAN and back-end SAN are generally separate, construction of two different networks is needed. On the contrary, in the case of a PC cluster consolidated with IP-SAN, both networks can be unified into a single commoditized network built with TCP/IP and Ethernet by using iSCSI. Therefore, the reduction of network construction cost and the increase in efficiency of operational management can be achieved.

However, in the PC cluster consolidated with IP-SAN, the bulk data transmitted for storage access and the commnication packets transmitted between nodes are intermingled with each other via the same network, which is based on TCP/IP over Ethernet. Therefore, compared with the separate-type networks, a load on the network should be concerned when a parallel and distributed application is executed. Thus, the PC cluster consolidated with IP-SAN needs to be evaluated how the integration affects performance, compared with PC cluster having separate-type networks.

3 Association rule mining and its parallelization

As an example of data-intensive applications, we have chosen one of data mining applications, called association rule mining. In association rule mining, in order to extract useful regularity and/or a useful relation from huge volumes of data, the frequency (support value) of appearing a certain pattern is examined. Such date-intensive applications are parallelized and executed on a PC cluster. Since the data processed by association rule mining is often huge, and a database is distributed in some cases, the applicaction program has been parallelized and implemented on a PC cluster. We have implemented the following two sorts of algorithm for association rule mining.

3.1 Apriori algorithm

Apriori algorithm was proposed by Agrawal and others in 1994, which generates a candidate itemset from the discovered frequent itemsets, and performs iterative numeration for association rule mining[5]. Since candidate itemsets should be generated in Apriori algorithm, there is a problem for which a mass of memory is needed when a database is scanned repeatedly.

Among some algorithms using Apirori as the base of parallelization, Hash Partitioned Apriori (HPA), which parallelizes Apriori with a hash function, is used in this paper.

3.2 FP-growth algorithm

FP-growth algorithm was proposed by Han and others in 2000[6]. FP-tree is used in FP-growth, which is a data structure in which compressed information required for association rule mining is stored compactly from the huge transaction database. This algorithm has an advantage over Apriori since a frequent pattern can be extracted without generating candidate itemsets. FP-growth discovers frequent itemsets using characteristics of FP-tree.

In this paper, Parallelized FP-growth (PFP) is used for the algorithm of parallel association rule mining of FP-growth[7]. FP-growth is said to be faster than Apriori. However, the FP-tree may become extremely large depending on the characteristics of data.

4 Experimental setup

In this experiment, multiple initiators access to a single target through Gigabit Ethernet in the PC cluster consolidated with IP-SAN. As target storage, SAS disks are used with RAID0 configuration, which is a common way of building high-performance storage in these days. Specification of each node of the cluster is shown in Table 1.

Table 1. Experimental setup：PCs

OS	initiator : Linux 2.6.18 target : Linux 2.6.18
CPU	initiator : Intel Xeon 3.6GHz target : quad-core Intel Xeon 1.6GHz
Main Memory	initiator : 4GB target : 2GB
HDD	initiator : 250GB SATA target : 73GB SAS × 2 (RAID 0)

Rocks [8] is used for the configuration and management of the cluster. Ganglia [9], which is a monitoring tool of the cluster, is also installed. As an initiator of iSCSI implementation, Open-iSCSI ver.2.0-865 is used [10]. As target software, iSCSI Enterprise Target (IET) ver.0.4.15 is used [11].

5 Experimental result and discussion

5.1 Bonnie++

First, we have evaluated the experimental system in which a heavy load is burdened on a network and a target by accessing a single target with multiple initiators. We have measured the sequential read and write accesses by using Bonnie++, a hard disk bench mark tool[12], to local device SAS disks (RAID0 configration) and the SAS disks as iSCSI target storage.

Figure 4 shows the total throughput of the storage access. Network traffic, CPU usage, and storage I/O of the target monitored by Ganglia, in the cases of one to eight initiators being connected to a target, are shown in Figure 5, 6, and 7, respectively.

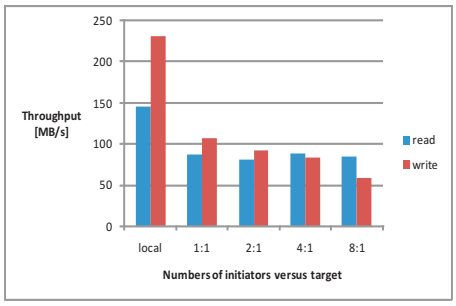

Figure 4. Sequential read access and Sequential write access

According to Figure 4, since local write access to storage is extremely fast with the power of RAID0 SAS disks, performance of iSCSI write access is below half of that of local write. On the other hand, performance of iSCSI read access is about 2/3 of that of local read. Thus in total, performance of iSCSI is considered to be relatively good, even compared with fast local storage. As the number of initiators increases, the performance of iSCSI read access remains almost the same, whereas that of iSCSI write access decreases due to the overhead of multiple accesses. The bottleneck of the system should be investigated further.

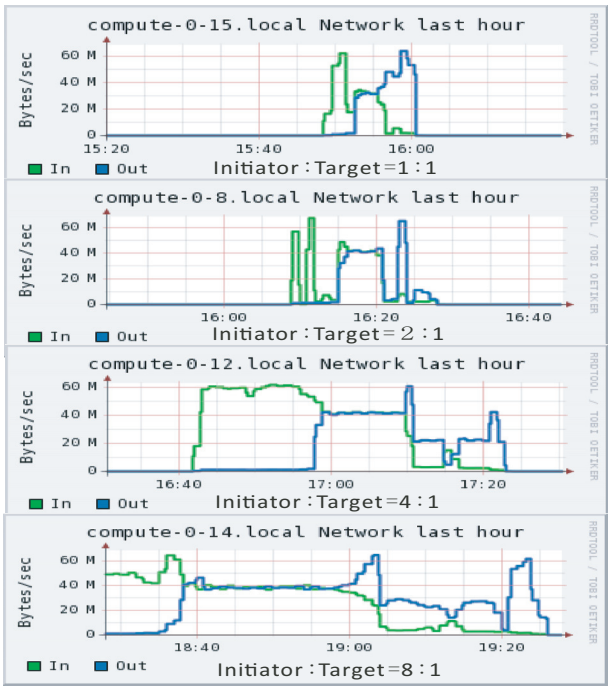

Figure 5. Behavior of network traffic of target during bonnie++ execution

As shown in Figure 5, network throughput of the target is 60Mbytes/sec (about 500Mbps) at most even in the case of eight initiators being connected to a single target. Because Gigabit Ethernet is used in this experimental system, there is still a margin in the network bandwidth.

By monitoring CPU usage shown in Figure 6, even though all initiators access to a single target, large percentage of CPU usage at target is " WAIT CPU ", which should be waiting time of I/O response. Therefore, load average of CPU is not high.

Compared monitoring result of storage I/O in Figure 7 with above observations, the bottleneck of this experiment should be storage I/O of the system. That is to say, performance of the PC cluster consolidated with IP-SAN, especially throughput of sequential write access, may be limited by the access speed of target storage.

However, the throughput degrades only a little even in such a case. Therefore, I/O performance of PC cluster consolidated with IP-SAN is considerably high.

5.2 Parallel data mining

Next, we have evaluated the PC cluster consolidated with IP-SAN using real applications, in which a single target is accessed by multiple initiators. HPA and PFP programs are executed on the same environment as previous Bonnie++

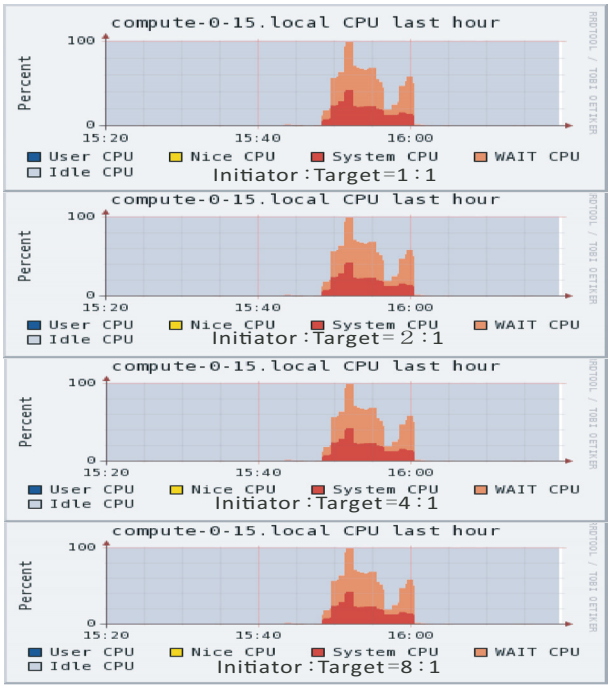

Figure 6. Behavior of CPU usage of target during bonnie++ execution

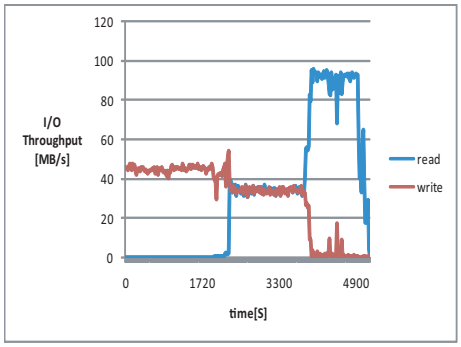

Figure 7. Behavior of storage I/O of target during bonnie++ execution (Initiator:Target = 8 : 1)

experiment.

Figure 8 shows the execution time when the HPA program is executed, and Figure 9 shows the case when the PFP algorithm is executed, when one to eight initiators are connected to a target. The number of items is 1000 in all cases, and the number of transactions is changed from 1M to 8M in the experiment.

As a result, the execution time hardly changes in any cases even when the ratio of the numbers of initiators and target changes, in both HPA and PFP cases.

By monitoring network traffic, there is still a margin in the bandwidth of the network. Although the storage is accessed with iSCSI, its traffic does not have a large influence on the network. Even if the back-end network is integrated into that of front-end, the bandwidth of the network is not consumed completely.

In the case of HPA, CPU utilization of initiator rises

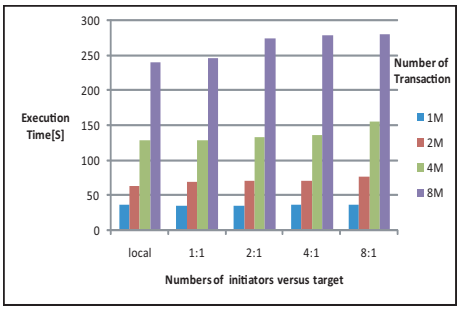

Figure 8. execution time when the HPA program is executed

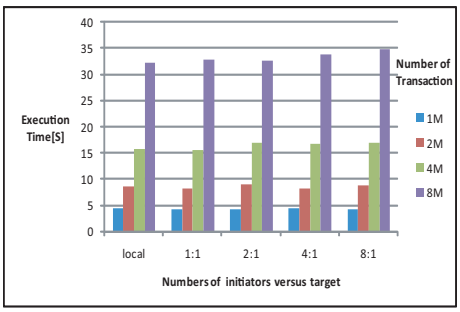

Figure 9. execution time when the PFP program is executed

repeatedly because this is iterative algorithm so that intensive computation is performed at each phase. Whole database is scanned repeatedly also at each phase, which should be the load to the network when iSCSI is used. However, the conflict of communication among nodes and storage access traffic on the network has degraded the total performance only little, even in the 8M transaction case.

In the case of PFP, CPU is not consumed so much because the computational complexity is low, although the FP-tree structure consumes much memory. In addition, database is scanned only twice during the execution. As a result, total performance is almost the same in all cases since the conflict of communication on the network should be low.

According to these results, a PC cluster consolidated with IP-SAN is considered to be practical for real data-intensive applications. Even with heavy access to iSCSI storage, total performance of the system remains almost the same.

5.3 multiple processes

Next, we have evaluated the PC cluster consolidated with IP-SAN using multiple processes, in which a single target is accessed by multiple initiators. HPA and PFP programs are executed at the same time bonnie++ is executed, on the same environment as previous experiments.

Figure 10 shows the execution time when the HPA program is executed while bonnie++ is executed, and Figure 11 shows the case when the PFP program is executed while bonnie++ is executed. The number of items is 1000 in all cases, and the number of transactions is changed from

1M to 8M.

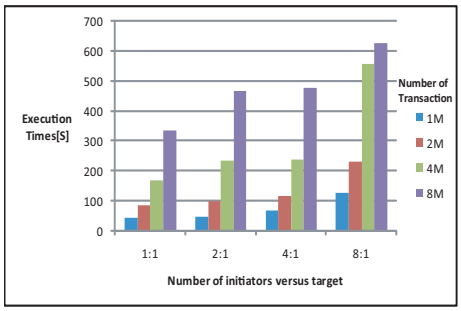

Figure 10. execution time when the HPA program and bonnie++ are executed

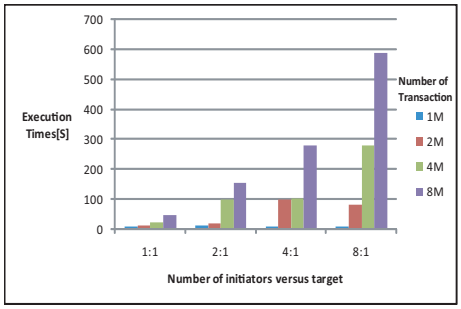

Figure 11. execution time when the PFP program and bonnie++ are executed

Different from the result of previous experiments, the execution time is long when a heavy load is burdened in both HPA and PFP cases. Moreover, the execution time becomes longer as the number of initiators connected to a target becomes large.

Figure 12 shows the network traffic of target during HPA and bonnie++ execution. According to this result, as the same with the previous experiment, network throughput of the target is 60Mbytes/sec (about 500Mbps) at most even in the case of eight initiators being connected to a single target. Because Gigabit Ethernet is used in this experimental system, there is still a margin in the network bandwidth. Thus, even in the case of heavy load by executing the storage access and data mining at the same time, performance of the network in the cluster is influenced only little.

Figure 13 shows CPU usage of initiator, Figure 14 shows CPU usage of Target, and Figure 15 shows storage I/O of target, respectively. When a single initiator is connected to a target, as Figure 13 shows, the ”USER CPU” is high at CPU usage of initiator because the storage access and data mining are busily executed. The state of CPU at the target seems to be waiting for I/O response according to Figure 14.

On the other hand, when the number of initiators connected to a target increases, CPU usage of initiator is ”WAIT CPU” mostly. As shown in Figure 15, the load to storage becomes high in this case.

According to these results, when the storage access is

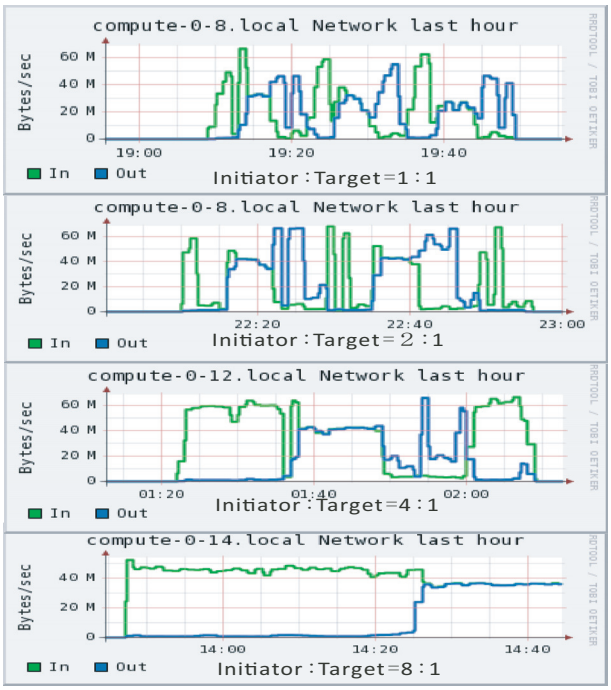

Figure 12. Behavior of network traffic of target during bonnie++ and HPA execution

light at the target, performance of application is bounded by CPU at initiators. However, as the storage access concentrates on the target, the total performance of the system is bounded by storage access, whereas it is not affected by the network virtually.

6 Conclusion

In this paper, a PC cluster consolidated with IP-SAN is evaluated when the network and the target are heavily burdened by accessing a single target from multiple initiators. A hard disk benchmark, data mining applications, and both of them are executed for the evaluation. In addition to measuring the execution time, network traffic, CPU usage, and storage I/O are monitored during the execution of the applications.

According to the result of the hard disk benchmark, throughput of sequential read access remains almost the same even when a heavy load is burdened on the network and the target. On the other hand, throughput of sequential write access decreases a little when the target is accessed by multiple initiators. The cause of this performance degradation seems to be storage I/O, whereas no bottleneck exists at the network nor CPU. Nevertheless, I/O performance of a PC cluster consolidated with IP-SAN is considerably high.

In the case of executing parallel data mining, there should be a collision on the network between the traffic of iSCSI storage access and the communication packets transmitted between nodes. However, the performance is almost the same even when the target is accessed by multiple initiators. In data-intensive applications, because not only storage is accessed but also intensive calculation is

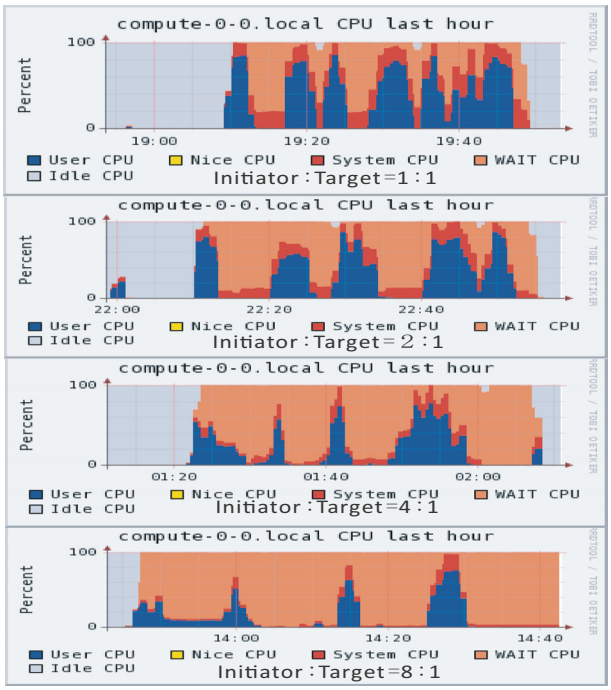

Figure 13. Behavior of CPU usage of initiator during HPA program and bonnie++ execution

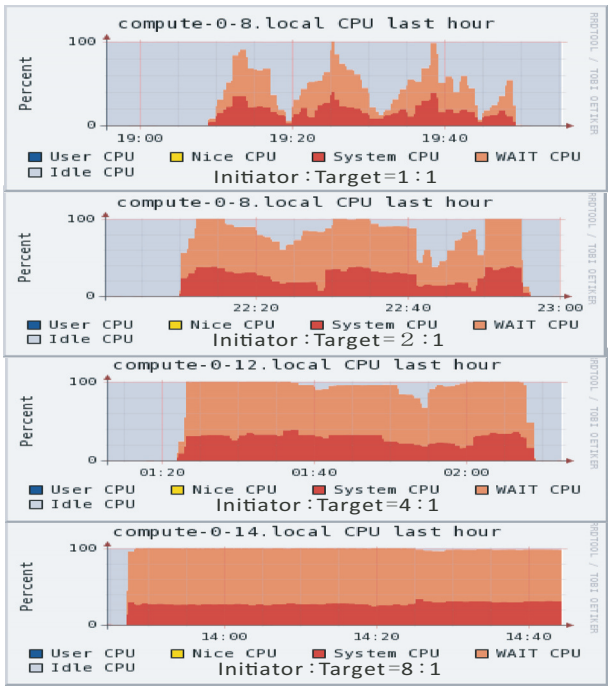

Figure 14. Behavior of CPU usage of target during HPA program and bonnie++ execution

performed at each node, there should be less collision on the network. Thus a PC cluster consolidated with IP-SAN does not degrade its performance when data-intensive applications are executed.

In the case of executing parallel data mining when a hard disk benchmark is executed, the execution time becomes longer than that of executing parallel data mining only, especially when a load of storage is high. Total performance of the system is CPU-bound or I/O-bound, not network-bound in this case.

As a future work, we will investigate the behavior of inside of the system. For example, by introducing a kernel monitor of OS, the detailed behavior of IP-SAN should be clarified. We will optimize the system using such a result.

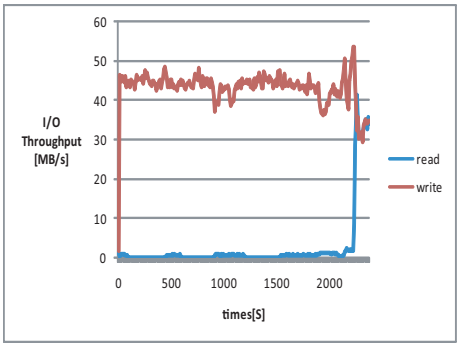

Figure 15. Behavior of storage I/O of target during HPA program and bonnie++ execution (Initiator:Target = 8 : 1)

References

[1] iSCSI RFC: http://www.ietf.org/rfc/rfc3722.txt

[2] D. Xinidis, A, Bilas, and M. D. Flouris, “Performance Evaluation of Commodity iSCSI-based Storage Systems,” MSST2005, April 2005.

[3] A. Joglekar, M. E. Kounavis, and F. L. Berry, “A Scalable and High Performance Software iSCSI Implementation,” USENIX FAST 2005, pp.267-280, December 2005.

[4] B. K. Kancherla, G. M. Narayan, and K. Gopinath, “Performance Evaluation of Multiple TCP connections in iSCSI,” MSST2007, September 2007.

[5] R. Agrawal, T. Imielinski, A. Swami: “Mining Association Rules Mining between Sets of Items in Large Databades,” VLDB1994, pp.487-499, September 1994.

[6] J. Han, J. Pei, and Y. Yin: “Mining Frequent Patterns without Candidate Generation,” ACM SIGMOD2000, pp.1-12, May 2000.

[7] Iko Pramudiono and Masaru Kitsuregawa: “Tree structure based Parallel Frequent Pattern Mining on PC cluster,” DEXA2003, pp.537-539, September 2003.

[8] Rocks Cluster: http://www.rocksclusters.org/

[9] Ganglia Monitoring System: http://www.ganglia.info/

[10] Open-iSCSI: http://www.open-iscsi.org/

[11] iSCSI-Enterprise Target: http://sourceforge.net/projects/iscsitarget/

[12] Bonnie++: http://www.coker.com.au/bonnie++/