Cache Effect for Power Savings of Large Storage Systems with OLTP Applications

Norifumi Nishikawa, Miyuki Nakano, and Masaru Kitsuregawa

Institute of Industrial Science, The University of Tokyo,
4-6-1 Komaba Meguro-ku, Tokyo 153-8505, Japan
{norifumi,miyuki,kitsure}@tkl.iis.u-tokyo.ac.jp
http://www.tkl.iis.u-tokyo.ac.jp/top/

Abstract. The power consumption of modern datacenters is increasing rapidly. Storage in datacenter consumes much power. Today, databases, especially those for OLTP, have become a major storage application in datacenters. Therefore, power-saving management for OLTP applications has become an important task for user budgets and datacenter operations. This paper presents a description of a novel power-saving method for large storage systems based on application I/O behavior of OLTP applications. Features of our approach are (i) measurement of actual RAID storage power consumption, (ii) analysis of I/O behavior characteristics of OLTP applications, and (iii) delayed write operation at a storage cache level based on the I/O behavior of OLTP applications. We present a measured result of power consumption of storages during an OLTP application is running, and simulations results of our power-saving methods with varying cache size of storage, which demonstrate that our method provides substantially lower storage power consumption than that of a conventional OLTP environment.

Keywords: Storage, Energy, Power saving, OLTP, Datacenter.

1 Introduction

Storages and servers aggregation at datacenters have increased datacenters' power consumption. The power consumption of servers and datacenters in the United States is expected to double during 2006 to 2011[1]. Especially, storages consume large quantities of power at large datacenters since the amount of digital data stored and managed at datacenters is increased rapidly as described in [6]. Thus, disk storage power-saving has become a major issue at datacenters[2,3].

Database Management Systems (DBMSs) are reported as major storage applications at datacenters[15]. Storage capacity shipments for DBMS account for more than 60% of all shipments of high-end class storage installations. Shipments for online transaction processing (OLTP) applications such as Enterprise Resource Planning and Customer Relationship Management constitute more than half of the shipments of storage installations for DBMS. Therefore, storages for OLTP applications are expected to be a major power consumption need

S. Kikuchi et al. (Eds.): DNIS 2011, LNCS 7108, pp. 256–269, 2011.

at datacenters. Reducing the power consumption of storage devices for OLTP applications is an important task that must be undertaken to decrease the power consumption of datacenters.

In the past few years, several studies have addressed these problems. The features of these studies are an estimation of I/O-issued timing by analyzing a source code of a transaction[19,17]. If a transaction execution time is enough longer than the time length of turning on/off a storage, we may reduce the storage power consumption easily. But if a transaction execution time is shorter than the time length of turning on/off a storage, it is difficult to decide the timing of turning off the storage. Generally, a transaction execution time of OLTP application is less than a few seconds, that is much shorter than a time length of turning on/off a storage. Therefore, it is difficult to apply these approaches to storages used by OLTP applications. In order to develop a power-saving method for the storages, understanding the detailed characteristics of I/O behavior of the OLTP application at runtime is important. However, no report describes the power consumption of an actual OLTP application running on a large RAID storage. We measure the power consumption characteristics of a storage actually, and analyze I/O behavior of a TPC-C application. Here, the TPC-C application takes as a benchmark program to represent OLTP applications[5]. We propose a novel power-saving method based on the I/O behavior characteristics of TPC-C applications.

The contribution of this paper is that we measured the actual power consumption of OLTP applications on RAID storage in detail using a power meter. The RAID storage we used is an Adaptive Modular Storage 2500 (Hitachi Ltd.). Another contribution of this paper is to propose a new power-saving method offering only slight OLTP performance degradation by considering TPC-C I/O behaviors. A salient feature of our approach is to reduce a storage power consumption by analyzing an I/O behavior of OLTP application and by controlling a storage cache policy based on the I/O behavior. Detailed analysis of I/O behavior while varying the cache size is reported. Modern datacenter's storages have hundreds of GB cache, we, therefore, utilize this cache for power saving of the storages. Finally, we evaluate our power saving approach with consideration of a cache effect. Our power-saving method enables reduction of storage power consumption by approximately 45% in the best case for active TPC-C applications in our simulation results.

2 Related Works

Today, many storage energy saving approaches have been proposed. Approaches described in [12,10,21] tries to enlarge I/O interval by using a cache memory of servers or storages. Other approaches described in [4,13,20,16] concentrate frequently accessed data into a small number of disks and turn off other disks. However, it is difficult to find less frequently accessed OLTP data without application level information because OLTP applications issue very high frequently random I/Os.

Application-aware power saving approaches are also proposed. A salient feature of the application-aware power saving approaches is that they acquire I/O timing and an I/O target disk drive from applications [11,7,9,17,8,14]. Therefore, these approaches show particular effectiveness for long-term applications such as scientific or batch applications. However, no report describes research that has tackled short-term transactions processing such as a TPC-C application.

3 Characteristics of Storage Power Consumption

3.1 Mearement Environment

Figure 1 presents an outline of storage used in power consumption measurements. The storage contains a controller that has two I/O processors (two cores each) and 2GB cache memories, and 10 units, which have 15 disk drives each. The disk drives in each unit constitute a RAID (13D+2P RAID 6). Disk drives in the units are 750 GB SATA 7200 rpm. The storage also has four power distribution units (PDU) which supply power to the controller and units. The controller and units have two power supply cables each. The voltage of each cable is 200 V. We connected two clamp sensors to each cable. The clamp sensor is connected to a power meter (Remote Measurement and Monitoring System 2300 Series; Hioki E.E. Corp.).

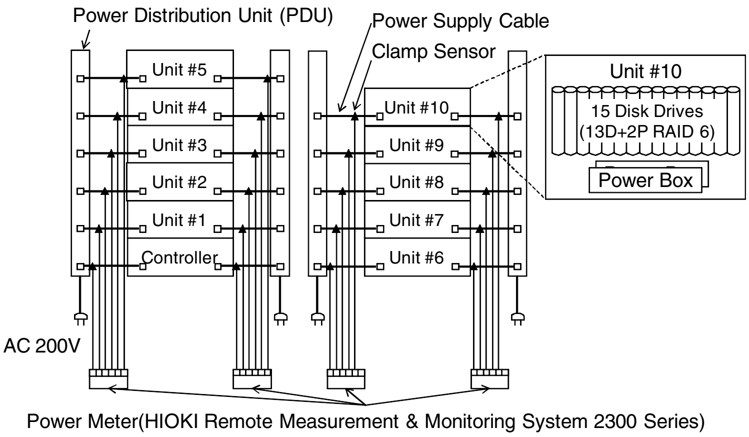

Fig. 1. Measurement Environment of Storage Power Consumption

Figure 2 portrays the system configuration used in our power consumption measurements. A load-generation server and the storage are connected by four 4-Gbit fiber channel cables. The server has 32 processors (2 cores each) and 512 GB memory. The OS of the server is AIX 5.3 64-bit version. The file system is JFS2. A capacity of one unit is 11.25TB and total capacity of the storage is approximately 112.5TB (both before constructing RAID). A capacity of storage cache is 2.0GB.

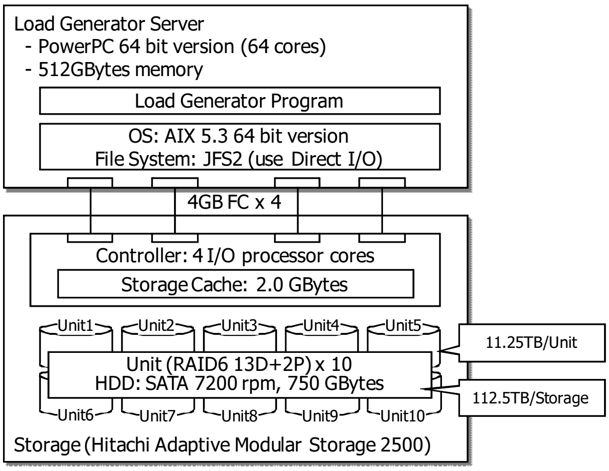
Load Generator Server
- PowerPC 64 bit version (64 cores)
- 512GBytes memory
Load Generator Program
OS: AIX 5.3 64 bit version
File System: JFS2 (use Direct I/O)
4GB FC x 4
Controller: 4 I/O processor cores
Storage Cache: 2.0 GBytes
Unit1
Unit2
Unit3
Unit4
Unit5
Unit (RAID6 13D+2P) x 10
HDD: SATA 7200 rpm, 750 GBytes
Unit6
Unit7
Unit8
Unit9
Unit10
11.25TB/Unit
112.5TB/Storage
Storage (Hitachi Adaptive Modular Storage 2500)

Fig. 2. System Configuration of Power Measurement Environment

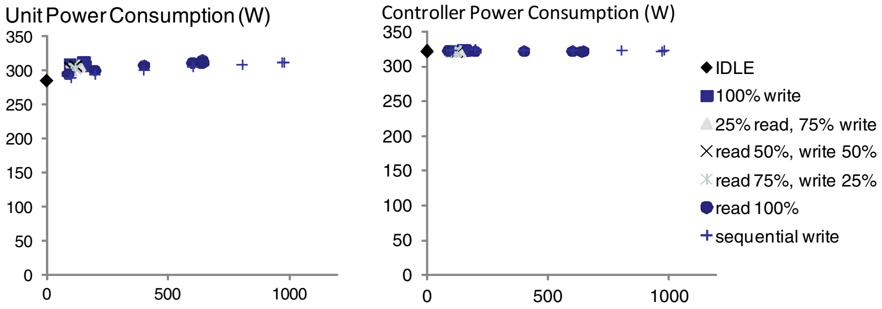

Fig. 3. Power Consumption and I/Os of the Unit (Left) and Controller (Right)

3.2 Power Consumption at an Active/Idle State

Figure 3 (left) depicts the relation between the unit power consumption and I/Os to units per second (IOPS). The I/O size is 8 KB. The figure shows that the power consumption of the unit increases slightly from idle status in accordance with the increase of IOPS. The maximum power consumption is about 315 W (+10.6% from the idle). Figure 3 (right) presents the relation between the controller power consumption and I/Os to the storage controller. As the figure shows, the power consumption of the controller is steady around 320 W.

3.3 Power Consumption of Spin-Down and Power off Modes

Figure 4 depicts the power consumption of the unit in idle, spin-down, and power-off states. The power consumption is decreased by 40.6% when the unit is

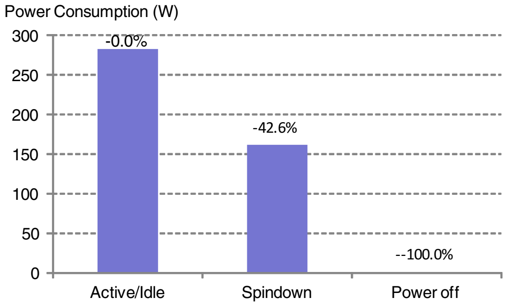

Fig. 4. Power Consumption of the Unit (idle, spin-down, and power off states)

in a spin-down mode. The power consumption is decreased by 100.0% when the unit is in a powered-off mode. The controller, on the other hand, cannot spin down or power off. Therefore, the power consumption of the controller is steady for all power statuses of the units.

4 Power Consumption and I/O Behavior of OLTP Applications

4.1 Experimental Environment

The hardware is the same configuration as that portrayed in Figs. 1 and 2. The software configuration is the following: the OS is a 64-bit version of AIX 5.3; the DBMS is a commercial DBMS for AIX; and the OLTP application is a tpcc-like program [18]. The file system cache is disabled (mounts with the direct I/O option). The DBMS buffer size is 25 GB maximum. The database is approximately 500 GB (the number of Warehouses is 5000), in which the Log data size is not included. With no actual report of power consumption of RAID storage and I/O trace of large OLTP application, we run the tpcc-like program and measured the results. We use 10 units and format them using the JFS2 file system. Log data are placed into Unit #1. All tables and indexes are placed into the other nine units (Units #2 - #10). Data of all tables and indexes are partitioned by hash into these units.

4.2 Power Consumption of TPC-C on RAID

Figure 5 presents the transition of power consumption of RAID storage at Figs. 1 and 2 while the tpcc-like program is running. The tpcc-like program was run for 30 min. The power consumption of the storage controller is steady at 319 W. The power consumption of unit #1 (for Log) is also a small fraction around 278 W. In contrast, the power consumption of units that contain database data was increased more than 10% compared to power consumption of the idle period. For detailed analyses, we used the 7 min of data included in the dashed line box in the figure.

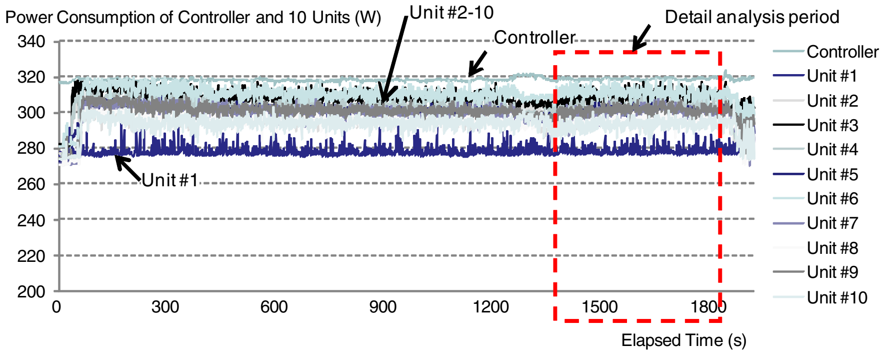

Fig. 5. Power Consumption of RAID Storage

4.3 I/O Behavior Characteristics of TPC-C

Results of Measurements. Figure 6 presents the number of reads and writes per second of each unit issued from the server. As presented there, the number of read I/Os is greater than that of write I/O. The IOPS to Log data (Unit #1) is higher than other units. The IOPS to tables and indexes (Unit #2 to #10) are almost equal among these units except Unit #2 and #10. The IOPS of these units are low because amounts of data in these units are fewer than other units (Unit #3 to Unit #9).

Figure 7 shows the quantities of reads and writes per second of each unit issued from the storage controller to disk drives in the units. As presented there, the quantities of read I/Os of units for tables and indexes are greater than those of write I/Os. The total number, however, is many more I/Os issued from the server. In contrast, the I/Os of unit for Log data are far fewer than I/Os issued from the server. Comparing the number of I/Os to storage controller (in Fig. 6), the number of I/Os issued to disk drives (in Fig. 7) are increased.

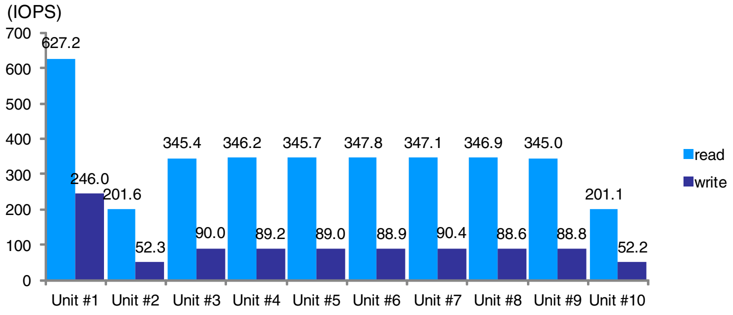

Fig. 6. Reads and Writes per second in a unit for TPC-C (Issued from DBMS to Storage Controller)

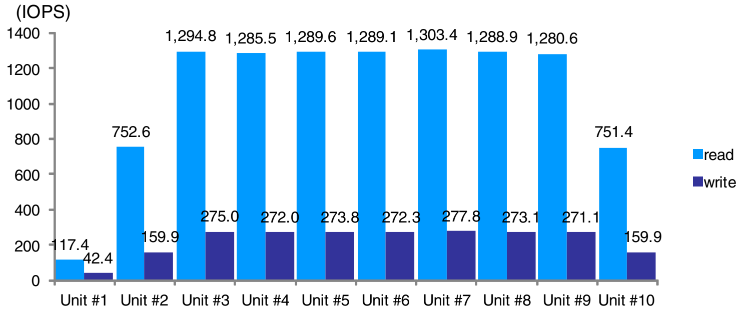

Fig. 7. Reads and Writes per second in a unit for TPC-C (Issued from Controller to Disk Drives of Units)

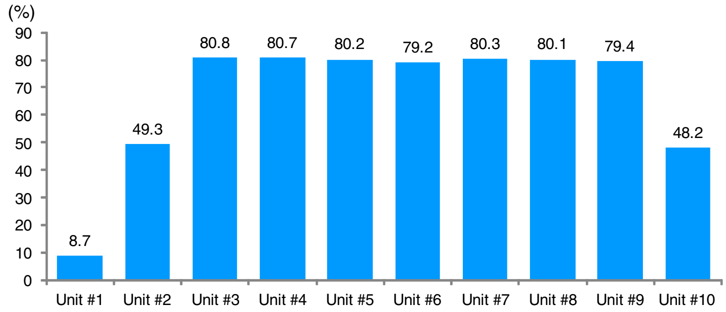

Fig. 8. Average Usage of Disk Drives

Figure 8 shows the average usage of 15 disk drives of each unit. As shown here, the disk drive usage of units for tables and indexes is more than 80% (without Unit #2 and #10). In contrast, the disk drive usage of unit #2 (for Log data) is only 8.7%.

Today's modern storages, on the other hand, have a large storage cache. Therefore we simulated the effect of storage cache in the case that the storage cache size is increased by using TPC-C I/O trace measured at this section. Figure 9 portrays the relation between the percentage of duplicated I/O and the percentage of storage cache size compared with the size of TPC-C database. Here, the duplicated I/O is an I/O which is issued to the same address that I/Os had been issued previously. As presented there, the rates of duplicated I/O are less than 1% when the cache size is less than 1%. The rates of duplicated I/Os, however, are increased rapidly where the storage cache size is larger than 1%. Figure 9 shows that a storage cache of only 5% size reduces more than 20% of I/Os to the disk drives of storage.

I/O Characteristics. I/O characteristics of TPC-C on RAID storage have the following characteristics:

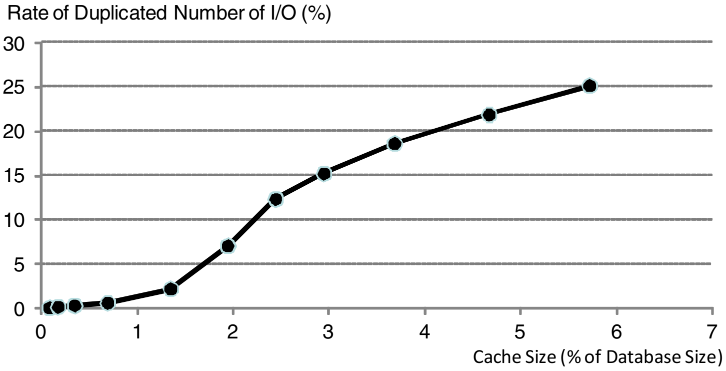

Fig. 9. Rate of Duplicated Number of I/O

1. I/Os are issued almost equally to units containing database tables and indexes (Units #2 to #10 in Fig. 6). This is because the data of all tables and indexes are partitioned into these units by hash.
2. The load of disk drives for Log data is low, and the loads of disk drives for tables and indexes are high (Fig. 8). The usage of disk drives for tables and indexes is greater than 80%, so these disk drives have little space to serve more I/Os.
3. Numbers of I/Os to disk drives containing database data are much higher than those of I/Os issued from the server, which results from calculation of the parity data of RAID 6 (Fig. 7). For one write from the server, the storage controller issues nine I/Os to disk drives (i.e., read old data and two old parity data, write new data and two new parity data. The storage also checks the written data by re-reading them because the SATA drive reliability is low). This overhead is known as a *write penalty.* The number of I/Os of Log data to disk drives are fewer than those of I/Os issued from the server. Most I/O of the Log data is sequential writes. Therefore, the storage controller merges multiple writes to one large write.
4. The writes to the same blocks is very low when the storage cache size is less than 1% of the database size. On the other hand, the writes to the same blocks becomes high for cases in which the capacity of storage cache is a few percent of the database size (see Fig. 9). This result illustrates that a small amount of storage cache reduces the load of disk drives on the storage effectively for TPC-C databases.

5 Power Saving Method for TPC-C Applications

We propose a power-saving method using the characteristics of I/O behaviors of TPC-C applications. To limit the power consumption of the storage, we use the locality of TPC-C application's I/O to disk drives in the storage. The main idea of our proposed method is a reduction of the *write penalty* by absorbing write

I/Os to the same blocks by using a storage cache, consolidation of RAID groups which store database data into a small number of RAID groups, and spin down or power off the remainder of RAID groups. We propose two simple approaches to reduce the number of I/Os to disk drives.

5.1 Allocation of Appropriate Size for Storage Cache

We described that write I/Os to the same blocks of TPC-C application is very low when the storage cache size is smaller than 1% of the TPC-C database size. On the other hand, the storage cache with size of only a few percent of the TPC-C database can reduce the number of I/Os to disk drives (shown in Fig. 9). Our first proposed approach is to allocate the appropriate size for the storage cache and reduce read I/Os to disk drives. The storage cache size of our experimental environment is 2 GB, but the maximum storage cache size of modern storage used at datacenters is hundreds of GB. The main usage of storage cache is to calculate parity data for RAID, to improve read response time by using read locality, and to improve write response time by using write back cache control method. Our proposed method uses the storage cache as an I/O buffer of DBMS with LRU cache replacement policy.

5.2 Write Delay of Storage Cache

The second approach for reducing I/O load of disk drives is a write delay of storage cache. The majority of I/Os to disk drives are a *write penalty*. Therefore, reducing the write penalty of disk drives is useful for the storage consolidation method. We propose a storage cache write delay method that maintains a constant amount of write I/Os into the storage cache and writes to disk drives when the number of dirty blocks reaches a threshold. Here, we select the threshold value as possible as large of storage cache except space required for parity data generation. We expect that this method reduces the number of blocks to be written to the disk drives, while also decreasing the write penalty.

6 Evaluation

To evaluate our proposed methods, we first calculated how many I/Os issued to disk drives would be decreased using our proposed methods. We then estimated the power consumption of the storage using the calculated number of I/Os. We simulated the number of I/Os by varying the storage cache using TPC-C I/O trace data measured in section 4. The experimental environment is the same as Fig 2.

6.1 I/O Rate Reduction

For the simulation of I/O rate reduction, we varied the size of storage cache to 0.4%, 1.0%, 3.0%, 5.0%, 10.0%, and 20.0% of the database size. We also varied the rate of dirty page to 1%, 10%, 25%, 75%, and 95% of the storage cache size. A flush of dirty blocks in the storage cache is delayed until the number of dirty blocks exceeds this dirty block rate (write delay). Multiple updates to the same block are merged to only one write to a disk drive.

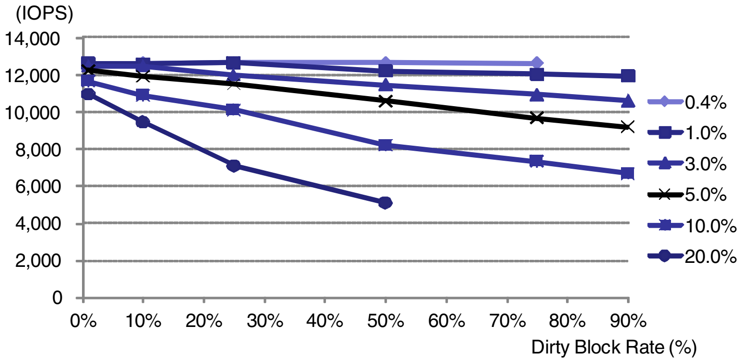

Fig. 10. Number of I/Os to Disk Drives when Changing the Cache Size and Dirty Block Rate

Figure 10 shows that the number of I/Os issued to disk drives is decreased according to the increase of the size of storage cache and the rate of dirty blocks. There is little effect when the storage cache is 0.4% of the TPC-C database size. On the other hand, a storage cache which has only a 3% size of a TPC-C database reduces the number of I/Os to disk drives more than 13% when the rate of dirty blocks is larger than 75% of storage cache size. When the storage cache size is 5% of TPC-C database size, the I/Os are decreased more than 23%. When the storage cache size is 20% of the TPC-C database size, the I/Os are decreased approximately 60%. These results show that a storage cache which has a few percentage size of the TPC-C database with a write delay has the capability of consolidating nine RAID groups to eight or seven RAID groups with little degradation of transaction throughput.

6.2 RAID Group Consolidation and Its Power Saving Effect

RAID Group Consolidation. Based on the preceding discussion, we calculate the necessary number of RAID groups to serve I/Os presented in Fig. 10. To calculate the number of RAID groups, we used a number of I/Os of 75% dirty block rate of each storage cache size (for 20% storage cache size, we used 50% dirty block rate). The number of I/Os that one RAID group can serve is 1,430 IOPS which are measured values shown in section 4. Table 1 shows the relation among storage cache size, number of I/Os to be served by storage, and the number of RAID groups to serve the required number of I/Os.

Table 1. Storage Cache Size and Number of RAID Groups

Storage Cache Size (%)	Number of I/Os to be Served (IOPS)	Number of RAID Groups
0.4	12,657	9
1.0	12,065	9
3.0	10,954	8
5.0	9,690	7
10.0	7,365	6
20.0	5,120	4

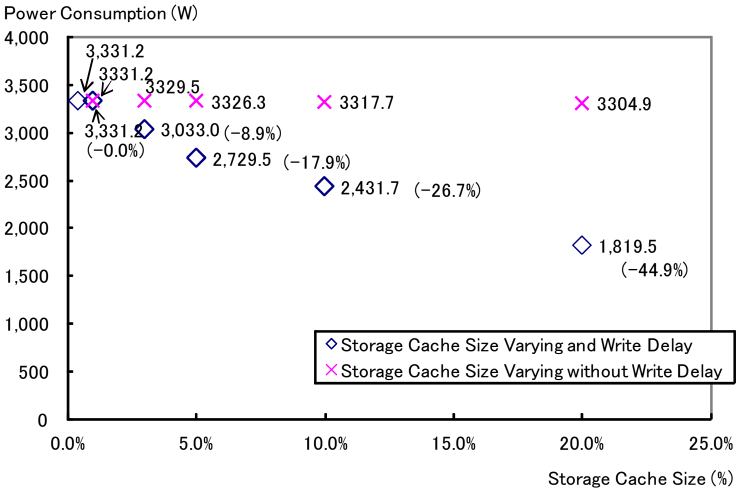

Fig. 11. Power Consumption of Storage when Varying the Storage Cache Size

Storage Power Consumption and Transaction Throughput. We calculated the power consumption of storage applying only the storage cache size varying method, and the RAID group consolidation with storage cache size varying and write delay method. Figure 11 presents the results. Square marks show the power consumption of the RAID Group consolidation method. Cross marks show the power consumption applying only the storage cache size varying method. Values in brackets show the reduction rate of power consumption of RAID Group consolidation method from the power consumption applying only the storage cache size varying method.

Figure 11 shows that the storage cache varying method does not reduce the power consumption of storage irrespective of the increment of storage cache. This is true because the increment of the storage cache size does not reduce the number of I/O to disk drives (1% line of Fig. 9). Therefore, reduction of the number of RAID groups is difficult without degradation of transaction throughput. The storage cache size varying method alone does not decrease the storage power consumption.

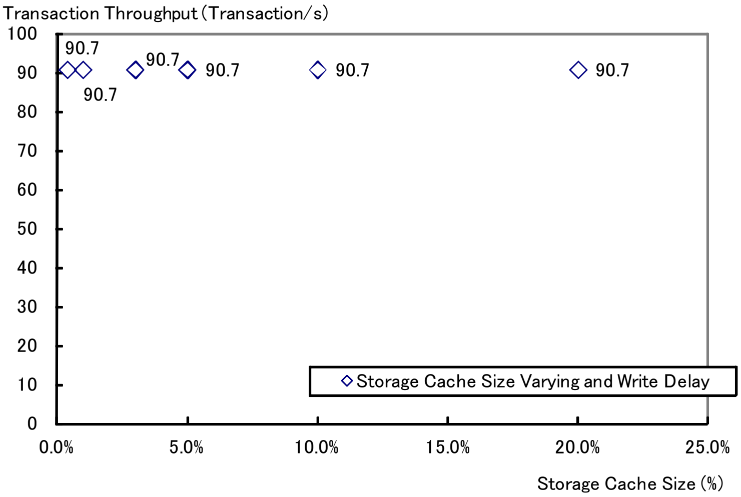

Fig. 12. Transaction Throughput when Varying the Storage Cache Size

However, RAID group consolidation using storage cache size varying and write delay method can reduce power consumption by 8.9% with 3.0% storage cache size, and can reduce power consumption by 17.9% using 5.0% storage cache size. The 5.0% size of the storage cache is approximately 20 GB, for the large TPC-C database with a scale factor such as 5000. The cache size of the high-end storage used at large datacenters can accommodate a storage cache much larger than 20 GB. Therefore, our proposed method is useful for large datacenters. We also calculated the power consumption of storage with cache sizes of 10% and 20%. Fig. 11 shows that the power consumption reduction rates are, respectively, 26.7% and 44.9%. Our proposed method has the capability of reducing the power consumption of storage used by active TPC-C applications drastically when the storage cache size is large.

Figure 12 depicts the calculated transaction throughput for each storage cache size. The transaction throughput is an important indicator. As Fig. 12 shows, the transaction throughput using our proposed method does not degrade the transaction throughput because our RAID group consolidation method using write delays controls the number of I/Os to disk drives to keep it from exceeding the number of I/Os portrayed in Fig. 7.

7 Conclusion

We measured the actual power consumption values of storage and considered the behavior of a TPC-C application in detail. We then proposed a novel power-saving method that reduces the power consumption of storage for TPC-C applications. The salient feature of our approach is the consolidation of the TPC-C database into a few RAID groups using appropriate storage cache size and a write delay method at the storage-cache level based on comprehensive behavior of OLTP DBMS executing multiple transactions. We demonstrated that our method achieves an approximately 45% reduction of the storage power consumption for a TPC-C application with little throughput degradation.

References

1. U.s. environmental protection agency energy star program, report to congress on server and data center energy efficiency public law, 109–431 (2007)
2. Bauer, R.: Building the green data center: Towards best practices and technical considerations (2008)
3. Chu, P.B., Reinsel, E.: Green storage ii: Metrics and measurement (2008)
4. Colarelli, D., Grunwald, D.: Massive arrays of idle disks for storage archives. In: ACM /IEEE 2002 Conference on Supercomputing, pp. 47–57 (2002)
5. Transaction Processing Performance Council. Tpc-c, an online transaction processing benchmark
6. Gens, F.: Idc predictions 2011: Welcome to the new mainstream. IDC White Paper #225878 (2010)
7. Gniady, C., Hu, Y.C., Lu, Y.H.: Program counter based techniques for dynamic power management. In: Proc. of 10th International Symposium on High Performance Computer Architecutre, pp. 24–35. IEEE (2004)
8. Harizopoulos, S., Shah, M.A., Meza, J., Ranganathan, P.: Energy efficiency: The new holy grail of data management systems research. In: 4th Biennial Conf. on Innovative Data Systems, pp. 112–123 (2009)
9. Heath, T., Pinheiro, E., Hom, J., Kremer, U., Bianchini, R.: Application transformations for power and performance-aware device management. In: 11th International Conference on Parallel Architectures and Compilation Techniques, pp. 121–130 (2002)
10. Li, D., Wang, J.: Eeraid: Power efficient redundant and inexpensive disk arrays. Proc. 11th Workshop on ACM SIGOPS European Workshop, 174–180 (2004)
11. Mandagere, N., Diehl, J., Du, D.H.-C.: Greenstor: Application-aided energy-efficient storage. In: MSST, pp. 16–29. IEEE Computer Society (2007)
12. Papathanasiou, A.E., Scott, M.L.: Energy efficient prefetching and caching. In: Proc. of USENIX 2004 Annual Technical Conference, pp. 255–268. USENIX Association Berkeley (2004)
13. Pinheiro, E., Bianchini, R.: Energy conservation techniques for disk array based servers. In: Proc. 18th Annual International Conference on Supercomputing, pp. 68–78. ACM (2004)
14. Poess, M., Nambiar, R.O.: Tuning servers, storage and database for power efficient data warehouse. In: 26th IEEE International Conf. on Data Engineering, pp. 1006–1017. IEEE Computer Society (2010)
15. Reinsel, D.: White paper datacenter ssds: Solid footing for growth. IDC White Paper # 210290 (2008)
16. Son, S.W., Chen, G., Kandemir, M.T.: Disk layout optimization for reducing energy consumption. In: ICS, pp. 274–283. ACM (2005)
17. Son, S.W., Kandemir, M., Choudhary, A.: Software-directed disk power management for scientific applications. In: Proc. of 19th IEEE International Parallel and Distributed Proceesing Symposium. IEEE Computer Society (2005)
18. Tkachenko, V.: tpcc-mysql

19. Ueno, Y., Goda, K., Kitsuregawa, M.: A study on disk array power reduction using query plan for database systems (2007)
20. Oldham Weddle, C., Qian, M.J., Wang, A.A.: Paraid: A gear-shifting power-aware raid. In: 5th USENIX Conference on File and Storage, pp. 245–267. USENIX Association (2007)
21. Yao, X., Wang, J.: Rimac: A novel redundancy based hierarchical cache architecture for power efficient. In: Proc. 2006 EuroSys Conference on High Performance Storage System, pp. 249–262 (2006)