2017 IEEE International Symposium on Parallel and Distributed Processing with Applications and 2017 IEEE International Conference on Ubiquitous Computing and Communications (ISPA/IUCC)

MCS-B:An Energy Efficient Storage System for Astronomical Observation Data Based on Logical Block Replacement Strategy

Chao Sun, Shanjiang Tang✉, Zichao Yuan, Ce Yu, Jian Xiao and Jizhou Sun
School of Computer Science and Technology
Tianjin University
Tianjin, China
Email: {sch, tashj, skyyuan, yuce, xiaojian, jzsun}@tju.edu.cn

Shuibing He
Computer School
Wuhan University
Hubei, China
Email: heshuibing@whu.edu.cn

***Abstract*—Power conservation for the storage system is crucial for cost saving and there are many approaches to address it, such as multi-level storage and data migration strategy. However, most of them are purely theoretical studies without real experiments. In realistic application scenarios, there can be a variety of factors that can impact the effectiveness of the energy efficient model. In this paper, a new energy efficient storage system for astronomical observation data called MCS-B is designed, which uses multi-level caching strategy and block replacement strategy. According to the manner of astronomical data access, MCS-B uses LB (logical blocks) to cluster correlated files and places them on the data disk via a temporal and spatial pattern. We further design a specialized prefetching and caching strategy to create larger disk idle time intervals so that the disks that are not in use within a certain time will be powered down. Experimental results show that MCS-B outperforms existing traditional energy efficient schemes in the energy consumption by 34.24% up to 42.39%. Moreover, the average response time for requests is reduced by 71.36% up to 77.14%.**

***Keywords*-energy efficient; controllable disk array; storage system; astronomical observation data; block replacement strategy**

I. Introduction

Nowadays the energy consumption of the storage system can't be neglected. In 2010, the energy consumption of data centers has accounted for 1% of total electricity consumption in China [1]. The power used of the data center falls into roughly three key categories: Server, Storage System and Network. Among various components of a data center, storage system is one of the biggest energy consumers, consuming almost 40% of the total [2]. Thus, how to reduce the energy consumption of the storage system is becoming one of the hottest researches.

In recent years, a wide variety of energy efficient architectures, algorithms, and methods of storage system which take too much consideration of generality have been proposed. They only presented generally theoretical models rather than available solutions for realistic application scenarios. Therefore, in some realistic application scenarios especially astronomical observation presented in this paper, almost all general-purpose technologies are not applicable, which makes customizing the specific application scenarios of energy storage methods become important.

Corresponding Author: Shanjiang Tang, tashj@tju.edu.cn

Astronomical observations generate a large amount of data, and every telescope has its own data format and data processing pipeline. China is establishing a series of telescopes for Antarctic astronomical observations named AST3. The AST3 project consists of three large field of view survey telescopes with 680mm primary mirror (500mm effective), mainly for observations of supernovas and extrasolar planets searching from Antarctic Dome A where is very likely to be the best astronomical site on earth for astronomical observations from optical wavelength to thermal infrared and beyond [3]. Each telescope of AST3 is equipped with a CCD camera of 10K x 10K, but in actually only a half used for observation and the other for read data. When in the sky survey mode, each of the AST3 telescopes theoretically provides 200MB images per 2.4 minutes which reaches up to 360GB a day.

Now, all observation data can only be fetched back home by the scientists at the end of the summer research and they process all the data domestically, which causes various astronomical phenomena missed due to the non-real-time data processing. So it is inevitable to establish a data center on Dome A. However, restricted by the power equipment, the peak-power limitation is 1kW, so the data center should be energy-efficient. And designing an energy-saving storage system to provide services for the storage of the observation data produced by the telescope and data processing services for the astronomers will be more effective in this scene.

In this paper, a new energy efficient storage system for astronomical observation data called MCS-B is designed. MCS-B uses multi-level caching strategy and block replacement technique. According to [4], astronomical data are usually used according to the temporal and spatial requirements. To boost the data disks' performance, MCS-B utilizes LB (logical block) to cluster correlated files and places them on the data disk via a temporal and spatial pattern. In order to realize the goal of energy efficiency,

0-7695-6329-5/17/31.00 ©2017 IEEE
DOI 10.1109/ISPA/IUCC.2017.00037

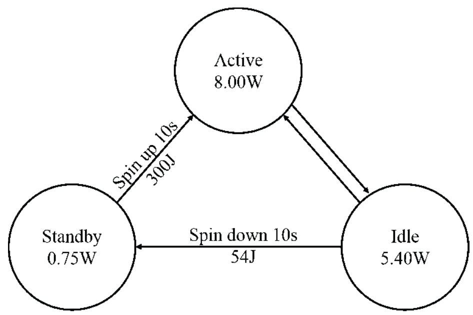


Figure 1. Seagate ST2000DM001 Energy Model

MCS-B also uses the prefetching and caching techniques to create larger disk idle time intervals. Since the disk array of the storage system is customized, each disk can be powered down and up.

More specifically, this paper makes the following contributions:

- MCS-B is specifically designed for Astronomical Observation Data of AST3. Furthermore, MCS-B has a strong adaptability, making it can be easily ported to other general-purpose telescope system.
- MCS-B uses multi-level caching strategy and block replacement technique to create larger disk idle time intervals.
- According to the usage of astronomical data, MCS-B utilizes LB to cluster correlated files and places them on the data disk via a temporal and spatial pattern.
- Compared with the file replacement strategy, MCS-B uses LB as the basic unit of the replacement strategy. This not only reduces the computation of the priority, but also reduces the frequent small erase operation and the disk fragmentation. It also improves the reliability of the system by introducing ECC (the Error Correcting Code) of the files on the block.
- The experimental results show that compared with existing energy efficient architectures, the energy consumption of MCS-B is reduced by 34.24% up to 42.39%, moreover, the average response time for requests is reduced by 71.36% up to 77.14%.

The rest of this paper is organized as follows. Section 2 presents the related works and the existing disk energy efficient strategies. Section 3 presents the basic hardware and system architectures of MCS-B, and its data distribution strategy. Section 4 presents the energy consuming analysis, LB migration strategy and the workload analysis of MCS-B. Section 5 evaluates the performance of the proposed energy efficient technique by comparing with existing approaches. Finally, the conclusions of this paper and the future work are presented in section 6.

II. Related Work

The Seagate ST2000DM001 energy model [5] is shown in Fig. 1. There are three states for the single hard disk: Active, Idle and Standby. The storage system saves energy by making idle disks into the standby state. The energy-saving theory of disk storage system is that if no data accessed, the disk will enter relatively low energy consumption status [6]. Hence, it is well known that energy savings can be improved by extending the expected length of the disk idle period. When the interval time of a disk exceed a limit time T_{BE}(a minimum interval that the disk enters standby state), the disk immediately enter the standby state [7] [8] [9].

Several methods have been used to extend the expected idle time of disks, such as Cache and Data Prefetching Strategy and Data Migration Strategy.

A. Cache and Data Prefetching Strategy

MAID (Massive Arrays of Idle Disks) was the first technique proposed to use a subset of disk drives as caches for a larger disk system and using data prefetching strategy [10]. MAID is divided into zero or more "cache drives" that constantly spinning; the remaining "data drives" are allowed to spin-down after a varying period of inactivity. With the emergence of new storage media, a lot of cache strategies based on a memory or SSD were proposed, such as: E-HASH [11], FLAP [12] and PLC-cache [13]. Meanwhile, a lot of data prefetching strategies were proposed, such as: PRE-BUD [14] [15] [16], Eco-Storage [17] and DATS [18].

B. Data Migration Strategy

PDC (Popular Data Concentration) [7] and PDC-NH (Popular Data Concentration on NAND Flash and Hard Disk Drive) [19] migrate frequently accessed data to a subset of the disks, so that other disks will enter the standby state. The idea behind PDC is to dynamically migrate the popular disk data (i.e., the most frequently accessed data on disk) to a subset of the disks in the array, so that the load becomes skewed towards a few of the disks and others can be sent to low-power modes. A lot of improvement program were proposed, such as: FDTM [20], LAM [21], EESDC [22] and EDM [23].

III. Basic Design of MCS-B

This section shows the basic design of MCS-B, including the basic hardware of Controllable Disk Array, system architectures of MCS-B and its data distribution strategy.

A. Controllable Disk Array

Motivated by strictly limited energy of the Antarctic and harsh environmental conditions, the disk of the storage system is customized.

The controllable disk array of the storage system is shown in Fig. 2. It comprises two parts: the industrial control computer and the disk array. The disk array consists of 40

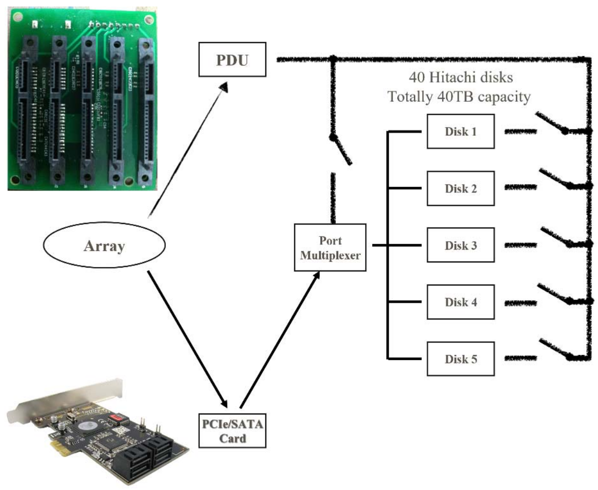


Figure 2. Controllable disk array of the storage system

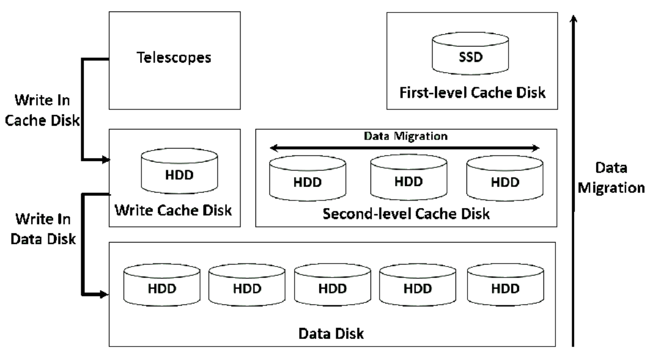


Figure 3. Basic architechture of MCS-B

2.5-inch mechanical hard disks which can be controlled by PDU (Power Distribution Unit) to power down or power up, so the disks just have two states: Active State and Closed State. Furthermore, the PDU can be ordered by instructions. Based on this, MCS-B can power down most of idle disks to reduce power consumption. The system should decide when to power down and power up the disks, what to do when the data are crashed, and how to rebuild the system. The reliability strategy of AST3 is detailedly introduced in our another paper [24].

B. System Architecture

As is shown in Fig. 3, MCS-B's basic architecture is the same as our previous work MCS-SSD's [25]. It uses cache strategy, and divides disks into data disks and cache disks. Furthermore, the cache disks are divided into two levels: the first-level cache disks uses SSD (solid state disks) to cache the most popular prefetching data, while the second-level cache disks are composed with HDD (hard disk drive).

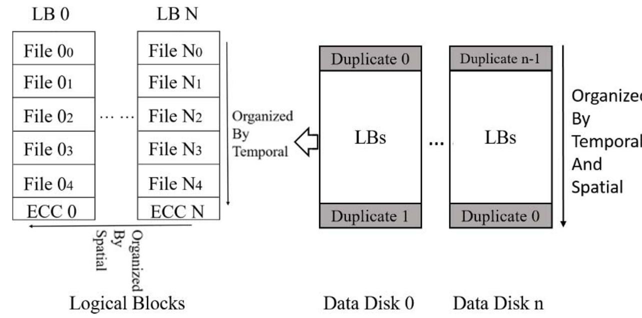


Figure 4. Data Distribution Strategy

C. Data Distribution Strategy

Astronomical data access is often associated with time and space. For example, in order to study new extrasolar planets or supernovas, it needs to compare the data observed in the same sky zone at different time to pick out sources with changed stars by light curves. Meanwhile, the size of the astronomical observation data is fixed, ranging from 150MB to 170MB. Hence, MCS-B uses logical blocks to cluster correlated files.

The LB's architecture is shown in Fig. 4. Compared with XB (Exchange Blocks) [26] and other block replacement techniques [27], LB's size is fixed. Hence, LB reduces the frequent small erase operation and the disk fragmentation to extend the lifetime of SSD. Moreover, LB has redundant space to record ECC [28] [29] of the files on LB. With ECC, the reliability of the storage is improved.

In order to realize the data distribution strategy, MCS-B uses DARAW [30] strategy to serve write requests of the telescope. DARAW is able to improve parallel I/O energy efficiency by the virtue of leveraging buffer disks to serve a majority of incoming write requests, thereby keeping data disks in low-power state for longer period times.

As shown in Figure 3, the data observed by the telescope directly is written into write cache disk group, then clustered in a LB. When the destination disk of the observation data is operational, it will write data to the corresponding data disk from the write cache disk. So writing operations of disks have a small effect on the energy saving of disks, MCS-B pays more attention to the reading operations.

Furthermore, as shown in Figure 4, in order to open less data disks when prefetching associated files, MCS-SSD puts some data duplicates at the edge of the adjacent disks. So when user requests the data on the edge, it only needs open one disk to prefetch associated LBs, reducing the number of open disks.

IV. Analysis

In AST3, there are three parts, i.e., the telescopes, the main control computer and the storage system, dominate

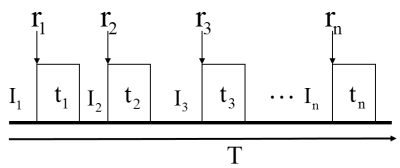


Figure 5. Request sequences of a disk within time T

the energy consumption. . The storage system is one of the biggest energy consumers, consuming about 1/3 of the total. Hence, an energy efficient storage system for AST3 is urgently required.

The work flow of the storage system is simple. When a user's requests arrive, the system searches data in the first level cache disk at first, requests that hit in the cache are sent to the user and the file's priority is modified; requests that miss in the first level cache disk are passed to the second level cache disk. If hit, the system sends the data to users, modifies the data's priority and moves the hit data and its associated data to the cache disk; otherwise requests are passed on to the data disk. If requests are hit in data disk, the system sends the data to users; if not, the system sends "not found" error message to the user.

A. Energy Consuming Analysis

The storage system saves energy by making idle disks into the standby state. But there are several challenges. One is not every idle interval of disks can be used, since the next access may arrive when the disk is shut down to the closed state, which will cost more energy to spin up the disk to the active state and greatly extend the request's response time.

Fig. 5 shows the request sequences of a disk within time T. In time T, the request sequences is $R = \{r_1, r_2, ..., r_n\}$, the idle interval is $I = \{I_1, I_2, ..., I_{n+1}\}$ and the service time is $t_{total} = \{t_1, t_2, ..., t_n\}$.

$$E = E_{active} + E_{idle} + E_{spinup} \tag{1}$$

Equation (1) is the energy consumption of a disk in time T. When the idle interval of the disk exceeds a limit time T_{BE}, the disk immediately enters the closed state, and if data needed, the disk will spin up for service. But the next interval is notoriously difficult to be predicated, so the perfect disk scheduling model is difficult to achieve. To solve this problem, a lot of strategies were proposed, such as dynamic power management [31] [32] [33] and DCAPS [34]. In the disk scheduling mode of the storage system based on MCS-B, it tries to predicate the ensuing interval, and then decides the next state of disks. The system also uses a cache strategy to increase the interval, so the disk may be under a higher probability to enter the standby state. In the storage system, it combines popular data queue and related data queue to prefetch data and it balances the load between overload disks and non-overload disks.

B. LB Migration Strategy

The LB migration strategy between the second-level cache disk and data disk is based on users' requests. MCS-B will move ALB (the LB which is accessed) and RLB (its associated LB) into the second-level cache disk. The amount of RLB is dynamic, and changed by the available space of the cache disks.

The LB migration strategy between the first-level cache disk and the second-level cache disk is based on the priority queue. Every second-level cache disk has its own priority queue (PQ1...PQN). The LBs on the top of the priority queue will be transferred to the first-level cache disk.

MCS-B uses the LRFU (Least Recently/Frequently Used) replacement policy to replace LBs in the cache disks. To describe the LB's priority, MCS-B have modeled the set of LBs as $LB = \{LB_1...LB_n\}$. And the LB_i is modeled as a set of parameters, such as the LB's size in Mbyte S_i, the number of LB's access R_i, the idle time of LB I_i, and the priority of LB P_i. Equation (2) implies that the priority increases by the number of LB_i's access (i.e., popularity) and decreases by the idle time of LB_i. k is a normalized parameter. The LB's priority will be computed after every request. If $P_i < P_{threshold}$ LB_i will be discarded from the cache disks.

$$P_i = \frac{k * R_i}{I_i} \tag{2}$$

C. Workload Analysis

In order to better illustrate the workload analysis, we first define the request's arrival rate. The request's arrival rate means the arrival rate of requests received by the storage system but excluding the requests introduced by the data migration. MCS-B divides the timeline into Time Windows of t seconds, then it can predict the request's arrival rate of the next time window. The predicted arrival rate (R) is calculated by (3), R_{former} is the request's arrival rate of the former time window, R_{mean} is the mean of all the previous request's arrival rate and w is the weight of the request's arrival rate of the former time window.

$$R = w \times R_{former} + (1 - w) \times R_{mean} \tag{3}$$

According to the value of R, MCS-B divides the workload into the light workload, the normal workload and the mixed workload. When R is small, it doesn't need so much cache disks, so MCS-B only uses the first-level cache disk. With the growth of R, the requests frequently arrive in a time window. The first-level cache disk cannot meet the users' requests, then MCS-B starts part of the second-level cache disks. MCS-B uses LB migration strategy on the second-level cache disks.

Table I
THE PARAMETERS OF THE SIMULATOR

Parameters Name	Value
SSD's Numbers	1
SSD's Total Size	1TB
Second-level Disks' Number	4
Second-level Disks' Total Size	8TB
Data Disks' Number	20
Data Disks' Total Size	40TB
Blocks' Number	29400
Block's Size	1000MB
Files in Block	5
Time Window	10s

In most instances, the workload of MCS-B is in dynamically changing workload conditions (Mixed Workload). With the prediction of R, MCS-B can change cache disks' status in advance. Hence, MCS-B is more suitable for the realistic application scenarios.

V. SIMULATION RESULTS

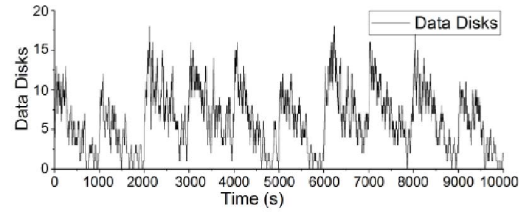
Before applying to the real storage system, a lot of simulation experiments of MCS-B are done on the simulator. Firstly, the simulator's architecture is described in subsection A. Combined with practical experience, MCS-B is fully tested in the light workload situation, the normal workload situation and the mixed workload situation. Subsection B, C, D and E will show the performance of MCS-B in these three situations. Finally, subsection F presents the conclusion of the simulations and compares the energy consumption of MAID-B (cache strategy of MAID with block replacement), which does not prefetch associated LBs to the cache disks.

A. Simulator Architecture

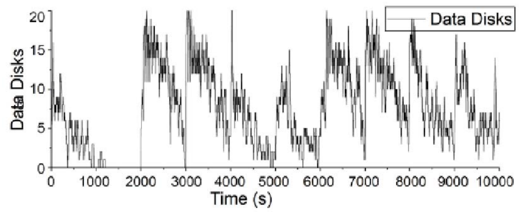
DiskSim is an efficient, accurate, highly-configurable disk system simulator [35], which is commonly used in the literature. However, the disk array used in the Antarctic is customized and the number of events needed to handle a file request is highly correlated, which makes DiskSim too slow for a realistic data center simulation that involves disks.

For performance evaluation, we design and implement a special simulator to simulate the storage system based on the real one. The simulator consists of 4 parts: requests generator, trace and statistics module, resource scheduling controller and environment builder. Requests generator impersonates user requests and generates the requests' queue. Trace and statistics module will trace the process of the simulation and record the statistical data. Resource scheduling controller processes the user's requests, and controls the storage system by the given energy saving strategies. The environment builder builds the basic conditions of the simulation, and consists of data builder and disk builder.

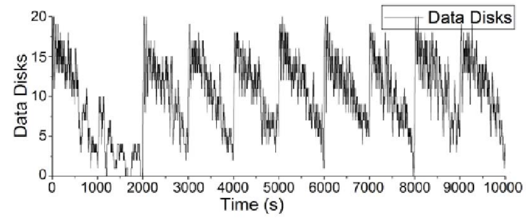
The simulator's parameters are shown in Table I. All modules of the simulator can be configured via changing parameters. The code of the simulator can be fetched at: https://github.com/yuanzichao/MCS-B,



(a) Light Workload



(b) Mixed Workload



(c) Normal Workload

Figure 6. The Data Disks' Status of All Situations

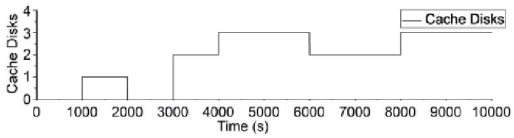
and the tracks of the workloads can be fetched at: https://github.com/yuanzichao/MCS-B/tree/master/track.

B. The Request's Arrival Rate (R)

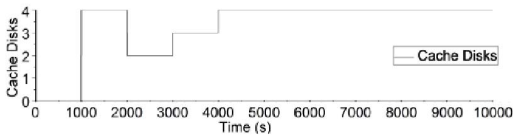
In light workload situation, the amount of requests in a Time Window ranges from 20 to 41, it doesn't need so much cache disks. While in normal workload situation, the amount of requests in a Time Window ranges from 97 to 114, then the first-level cache disk cannot meet the users' requests, so it will start part of the second-level cache disks to balance the workload of the first-level cache disk. The mixed workload situation's request's arrival amount in a Time Window ranges from 20 to 116, MCS-B will dynamically spin up and down the second-level cache disks by the request's arrival rate and the left space of the first-level cache disk. In most instances, the mixed workload is closer to the realistic application scenarios.

C. Data Disks' Status

The simulation results of MCS-B's data disks' status are shown in Fig. 6. According to the results, it shows that MCS-B's data disks' open numbers begin to decline after a short time and sometimes the data disks open numbers even can reach 0. That is because MCS-B prefetches LBs that associated with the hit LB to the cache disk. So after



(a) Mixed Workload



(b) Normal Workload

Figure 7. The Cache Disks' Status

some requests, the cache disks have stored a lot of LBs with high priority, most data disks can change to the closed state. This achieves a good energy saving effect.

The associated requests will last several time windows, so the cache will be invalid within a certain time, then the requests will be served by data disks. As shown in Fig. 6a, the request's arrival rate (R) is small in light workload situation, so the total open disks' amount is less than the mixed and normal workload situation. Furthermore, according to Fig. 6b and 6c, MCS-B can quickly adapt to different workload situation.

D. Cache Disks' Status

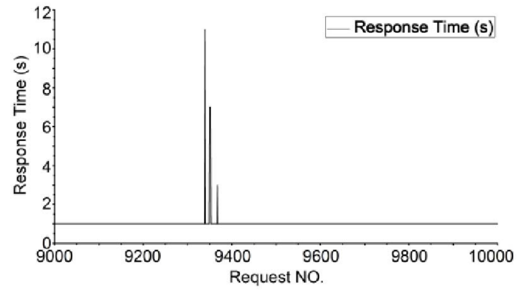
As shown in Fig. 7, the second-level cache disks are spined up and down by the request's arrival rate (R). At the beginning of the sequence of requests, MCS-B only opens the first level cache disk to prefetch LBs, but with the advent of more and more requests, the first level cache disk can't afford the users' requests, then the second level cache disks are opened. With the opening of the second level cache disks, more and more requests are responded from the second level cache disks, and the data disks have more chances to change to the closed state.

E. Requests' Response Time

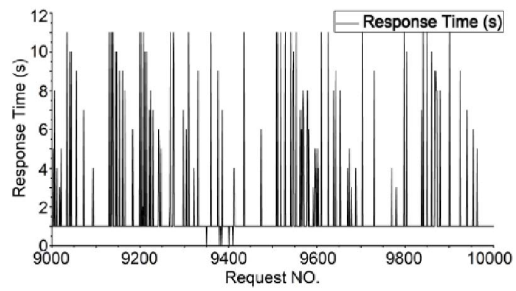
Fig. 8 shows the response time of the requests in the light workload situation, mixed workload situation and normal workload situation.

In the light workload situation, 90.21% (25478/28243) of requests are served by the first-level cache disk. Moreover, the Fig. 8a shows the response time of the request 9000-10000, it also proves that almost all requests are served by SSD. The average response time of all the 28243 requests is 2.03s, reduced about 71.36% than MAID-B's 7.08s.

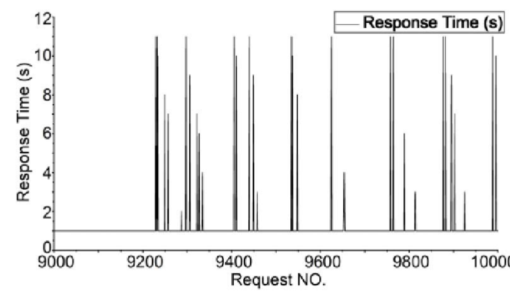
In the mixed workload situation, 85.87% (53226/61982) and 5.60% (3469/61982) of requests are served by the first-level cache disk and the second-level cache disks. Moreover, the Fig. 8b shows the response time of the request 9000-10000, the average response time of all the 103435 requests



(a) Light Workload



(b) Mixed Workload



(c) Normal Workload

Figure 8. Requests' Response Time of All Situations

is 1.78s, reduced about 75.44% than MAID-B's 7.26s. There are very few invalid requests, whose response time is 0s.

In the normal workload situation, 76.25% (78867/103435) and 16.43% (16994/103435) of requests are served by the first-level cache disk and the second-level cache disks. Moreover, the Fig. 8c shows the response time of the request 9000-10000, the average response time of all the 103435 requests is 1.65s, reduced about 77.14% than MAID-B's 7.23s.

According to Fig. 8, the cache and data prefetching strategies and the data migration strategy greatly enhance the quality of service and extend the data disks' idle time.

F. Conclusion Of Simulation

Table II shows the characteristics of the hard disk used in the simulation. The disk's power consumption is mainly comprised of two parts: the operating energy consumption and the spinning up energy consumption. The energy

Table II
THE CHARACTERISTICS OF THE HARD DISK

Description	Value
Disk model	Seagate ST2000DM001
Rotational speed	7200 rpm
Avg. seek time	8.5 msecs
Disk size	2000GB
Operating power	8.0 Watts
Spin up power	30 Watts
Spin up time	10 secs

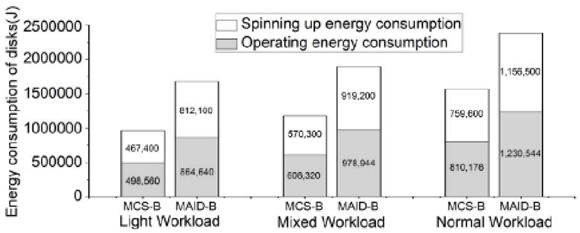


Figure 9. Energy consumption of MCS-B

consumption of MCS-B and MAID-B in light workload situation, mixed workload situation and normal workload situation is shown in Fig. 9. And the energy consumption of MCS-B's opened disks is about 42.39% less than MAID-B in light workload situations, 37.91% in mixed workload situations and 34.24% in normal workload situations. Furthermore, the average response time of the requests are respectively reduced by 71.36%, 75.44% and 77.14%. According to the energy consumption, the data disks' idle time are extended by MCS-B, so data disks have more chances to change to the closed state.

VI. CONCLUSION AND FUTURE WORK

In this paper, a new energy efficient storage system for astronomical observation data called MCS-B is designed, which uses multi-level caching strategy and block replacement strategy. MCS-B uses multi-level caching strategy and block replacement technique to create larger disk idle time intervals, so that the disks that are not in use within a certain time will be powered down. To boost the data disks' performance, MCS-B utilizes LB (logical block) to cluster correlated files and places them on the data disk via a temporal and spatial pattern by the usage of astronomical data. Compared with traditional file replacement strategy, MCS-B uses logical block as the basic unit of the replacement strategy. This not only reduces the computation of the priority, but also improves the reliability of the system by introducing ECC (the Error Correcting Code) of the files on block at the redundant space of the block. With the prediction of the request's arrival rate, MCS-B can better adapt to dynamically changing workload conditions. The experimental results show that compared with existing energy efficient architectures, the energy consumption of MCS-B is reduced by about 34.24% up to 42.39%, moreover, the average response time for requests is reduced by about 71.36% up to 77.14%.

This storage system based on MCS-B is also expected to play a significant role in the planning KDUST astronomical telescope project [36]. The observation data size will be larger and can reach petabyte (PB) level. Moreover, the system architecture and hardware architecture of KDUST astronomical telescope project are very complex. In future, we plan to extend our work so that it can support the KDUST.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (11573019, 61602336), the Joint Research Fund in Astronomy (U1531111) under cooperative agreement between the National Natural Science Foundation of China (NSFC) and Chinese Academy of Sciences (CAS).

REFERENCES

[1] L. Gu, F. Zhou, and M. Hui, "Research on energy consumption and energy efficiency of data center in our country," *Energy of China*, vol. 32, no. 11, 2010.

[2] J. Hiles, "Cost control and green initiatives," http://wwpi.com/cost-control-and-green-initiatives-4/, 2010.

[3] X. Yuan, X. Cui, B. Gu, S. Yang, F. Du, X. Li, D. Wang, X. Li, X. Gong, H. Wen *et al.*, "The ast3 project: Antarctic survey telescopes for dome a," in *SPIE Astronomical Telescopes+ Instrumentation*. International Society for Optics and Photonics, 2014, pp. 91 450F–91 450F.

[4] M. J. Graham, S. G. Djorgovski, A. Mahabal, C. Donalek, A. Drake, and G. Longo, "Data challenges of time domain astronomy," *Distributed and Parallel Databases*, vol. 30, no. 5-6, pp. 371–384, 2012.

[5] Seagate, "Desktop hdd product manual," http://www.seagate.com/www-content/product-content/barracuda-fam/desktop-hdd/barracuda-7200-14/en-us/docs/100686584p.pdf, 2015.

[6] X. Xie, "Research on key technologies of energy conservation for disk-based storage systems," 2012.

[7] E. Pinheiro and R. Bianchini, "Energy conservation techniques for disk array-based servers," in *25th Anniversary International Conference on Supercomputing Anniversary Volume*. ACM, 2014, pp. 369–379.

[8] F. Douglis, P. Krishnan, and B. Marsh, "Thwarting the power-hungry disk." in *USENIX Winter*, vol. 292, 1994, p. 306.

[9] C. Cunha, A. Bestavros, and M. Crovella, "Characteristics of www client-based traces," Technical Report BU-CS-95-010, Computer Science Department, Boston University, Tech. Rep., 1995.

[10] D. Colarelli and D. Grunwald, "Massive arrays of idle disks for storage archives," in *Proceedings of the 2002 ACM/IEEE conference on Supercomputing*. IEEE Computer Society Press, 2002, pp. 1–11.

[11] J. Hui, X. Ge, X. Huang, Y. Liu, and Q. Ran, "E-hash: an energy-efficient hybrid storage system composed of one ssd and multiple hdds," in *Advances in Swarm Intelligence*. Springer, 2012, pp. 527–534.

[12] L. Yang and W. Wang, "Flap: Flash-aware prefetching for improving ssd-based disk cache," *Journal of Networks*, vol. 9, no. 10, pp. 2766–2774, 2014.

[13] J. Liu, Y. Chai, X. Qin, and Y. Xiao, "Plc-cache: Endurable ssd cache for deduplication-based primary storage," in *Mass Storage Systems and Technologies (MSST), 2014 30th Symposium on*. IEEE, 2014, pp. 1–12.

[14] A. Manzanares, K. Bellam, and X. Qin, "A prefetching scheme for energy conservation in parallel disk systems," in *Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on*. IEEE, 2008, pp. 1–5.

[15] A. Manzanres, X. Ruan, S. Yin, M. Nijim, W. Luo, and X. Qin, "Energy-aware prefetching for parallel disk systems: Algorithms, models, and evaluation," in *Network Computing and Applications, 2009. NCA 2009. Eighth IEEE International Symposium on*. IEEE, 2009, pp. 90–97.

[16] A. Manzanares, X. Qin, X. Ruan, and S. Yin, "Pre-bud: Prefetching for energy-efficient parallel i/o systems with buffer disks," *ACM Transactions on Storage (TOS)*, vol. 7, no. 1, p. 3, 2011.

[17] M. M. Al Assaf, X. Jiang, M. R. Abid, and X. Qin, "Eco-storage: A hybrid storage system with energy-efficient informed prefetching," *Journal of Signal Processing Systems*, vol. 72, no. 3, pp. 165–180, 2013.

[18] M.-H. Kuo, H.-P. Chang, and D.-W. Chang, "A disk state-aware task scheduler with energy efficient prefetching and caching," in *Cloud and Green Computing (CGC), 2013 Third International Conference on*. IEEE, 2013, pp. 42–47.

[19] D. Lee and K. Koh, "Pdc-nh: Popular data concentration on nand flash and hard disk drive," in *Grid Computing, 2009 10th IEEE/ACM International Conference on*. IEEE, 2009, pp. 196–200.

[20] X. Zhao, Z. Li, and L. Zeng, "Fdtm: block level data migration policy in tiered storage system," in *Network and Parallel Computing*. Springer, 2010, pp. 76–90.

[21] G. Zhang, L. Chiu, C. Dickey, L. Liu, P. Muench, and S. Seshadri, "Automated lookahead data migration in ssd-enabled multi-tiered storage systems," in *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*. IEEE, 2010, pp. 1–6.

[22] Y. Chai, Z. Du, D. Bader, X. Qin *et al.*, "Efficient data migration to conserve energy in streaming media storage systems," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 23, no. 11, pp. 2081–2093, 2012.

[23] J. Ou, J. Shu, Y. Lu, L. Yi, and W. Wang, "Edm: An endurance-aware data migration scheme for load balancing in ssd storage clusters," in *Parallel and Distributed Processing Symposium, 2014 IEEE 28th International*. IEEE, 2014, pp. 787–796.

[24] C. Yu, L. Li, J. Sun, J. Xiao, J. Li, and Z. Shang, "A scalable real-time photometric system for automatic astronomical observations on dome a," in *Cluster, Cloud and Grid Computing (CCGrid), 2014 14th IEEE/ACM International Symposium on*. IEEE, 2014, pp. 900–905.

[25] Z. Yuan, C. Yu, J. Sun, J. Xiao, J. Wang, Z. Shang, and Y. Hu, "An energy efficient storage system for astronomical observation data on dome a," in *Algorithms and Architectures for Parallel Processing*. Springer, 2015, pp. 33–46.

[26] E. Otoo, D. Rotem, and S.-C. Tsao, "Workload-adaptive management of energy-smart disk storage systems," in *Cluster Computing and Workshops, 2009. CLUSTER'09. IEEE International Conference on*. IEEE, 2009, pp. 1–11.

[27] J.-W. Kim, "An efficient block replacement techniques for hybrid hard disk," *International Journal of Advancements in Computing Technology*, vol. 5, no. 12, p. 425, 2013.

[28] L. Zhang, W.-X. Gu, N. Liu, and W.-J. Wang, "Study and realization of error checking and correction algorithm based on the embedded solid state storage system," *Transaction of Beijing Institute of Technology*, vol. 35, no. 1, pp. 72–78, 2015.

[29] J. C. Chan, Q. Ding, P. P. Lee, and H. H. Chan, "Parity logging with reserved space: towards efficient updates and recovery in erasure-coded clustered storage." in *FAST*, 2014, pp. 163–176.

[30] X. Ruan, A. Manzanares, K. Bellam, X. Qin, and Z. Zong, "Daraw: A new write buffer to improve parallel i/o energy-efficiency," in *Proceedings of the 2009 ACM symposium on Applied Computing*. ACM, 2009, pp. 299–304.

[31] T. Bostoen, S. Mullender, and Y. Berbers, "Power-reduction techniques for data-center storage systems," *ACM Computing Surveys (CSUR)*, vol. 45, no. 3, p. 33, 2013.

[32] J. Chou, J. Kim, and D. Rotem, "Energy-aware scheduling in disk storage systems," in *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*. IEEE, 2011, pp. 423–433.

[33] T. Bostoen, J. Napper, S. Mullender, and Y. Berbers, "Minimizing energy dissipation in content distribution networks using dynamic power management," in *Cloud and Green Computing (CGC), 2013 Third International Conference on*. IEEE, 2013, pp. 203–210.

[34] M. Nijim, A. Manzanares, and X. Qin, "An adaptive energy-conserving strategy for parallel disk systems," in *Proceedings of the 2008 12th IEEE/ACM International Symposium on Distributed Simulation and Real-Time Applications*. IEEE Computer Society, 2008, pp. 75–82.

[35] J. S. Bucy, J. Schindler, S. W. Schlosser, and G. R. Ganger, "The disksim simulation environment version 4.0 reference manual (cmu-pdl-08-101)," *Parallel Data Laboratory*, p. 26, 2008.

[36] "Major telescopes added to highest observatory," http://www.laboratoryequipment.com/news/2010/09/major-telescopes-added-highest-observatory, 2010.