

Method and Device for Writing Stored Data into Storage Medium Based on Flash Memory

Technical Field

The present application relates to the field of computer and network technologies, and in particular, to a method and a device for writing data for storage into a flash memory-based storage medium.

Background

SSDs (Solid State Drives) that use a flash memory as the storage medium and adopt electronic read/write have superior read/write performance and have become mainstream hardware of storage for personal computers and servers. The storage medium of SSD is a NAND Flash cell, each bit of which has a physical characteristic that the value of the bit can be changed from 1 to 0 by charging the transistor, while its value can be reset from 0 to 1 only through erasing. The erasing operation is carried out in a unit of block.

The life of a NAND Flash cell is measured by a number of erasing and writing on the cell. When a block reaches this number of erasing and writing, it is impossible to use the block for storing data, and the block becomes a bad block. As the number of bad blocks increases, the service life of SSD decreases sharply.

Therefore, an FTL (Flash Translation Level) balanced erasing and writing algorithm or a file system layer having balanced erasing and writing functions is used in current technologies to achieve, as much as possible, balanced erasing and writing for all blocks on the same SSD so as to extend the service life of an individual SSD. Examples include the FTL layer dynamic balancing strategy, the FTL layer static balancing strategy, JAFF2 (Journaling Flash File System Version 2), YAFFS (Yet Another Flash File System), and the like.

For a system having a plurality of SSDs available, however, one or more SSDs could often be subjected to too many times of erasing and writing that are far more than those on one or more other SSDs. As a result, the overly used SSD is damaged prematurely. If no high availability solution is applied to data stored on the SSD, it is possible to result in data loss; for a system that adopts a high availability storage solution, the system stability would be weakened due to additional data migration.

Summary

In view of this, the present application provides a method for writing data for storage into a flash memory-based storage medium, the method being applied to a central control function module for performing writing control over at least two physical storage units, and the method comprising:

obtaining a total cumulative erasing count of all blocks in each of the physical storage units;
and

writing the data for storage into at least one physical storage unit having the lowest total cumulative erasing count among physical storage units satisfying a preset writing condition.

The present application further provides a device for writing data for storage into a flash memory-based storage medium, the device being applied to a central control function module for performing writing control over at least two physical storage units, and the device comprising:

a total-cumulative-erasing-count member configured to obtain a total cumulative erasing count of all blocks in each of the physical storage units; and

a physical-storage-unit member configured to write the data for storage into at least one physical storage unit having the lowest total cumulative erasing count among physical storage units satisfying a preset writing condition.

It can be seen from the above technical solution that in the embodiments of the present application, the central control function module selects, based on a total cumulative erasing count of all blocks in each physical storage unit, a physical storage unit having the lowest total cumulative erasing count from physical storage units satisfying a preset writing condition for writing of data for storage, thereby achieving balanced erasing and writing among different physical storage units, avoiding premature damage to a single physical storage unit due to too many times of erasing and writing on the physical storage unit, lowering the possibility of data loss in a non-high availability scenario, and improving the stability of a system in which the physical storage unit is located in a high availability scenario.

Brief Description of the Drawings

FIG. 1 is a schematic structural diagram of the composition of a host machine in a first application example according to some embodiments of the present application;

FIG. 2 is a schematic structural diagram of the composition of a cluster in a second application example according to some embodiments of the present application;

FIG. 3 is a flow chart of a method for writing data for storage according to Embodiment I of the present application;

FIG. 4 is a flow chart of a method for writing data for storage in a cluster according to Embodiment II of the present application;

FIG. 5 is a hardware structural diagram of the device according to some embodiments of the present application;

FIG. 6 is a logic structural diagram of a device for writing data for storage according to some embodiments of the present application;

FIG. 7 is a logic structural diagram of a cluster according to some embodiments of the present application.

Detailed Description

The embodiments of the present application provide a novel method for writing data for storage into a flash memory-based storage medium, the method comprising: selecting, based on statistical data of a total cumulative erasing count of all blocks in each physical storage unit, one or more physical storage units having the lowest total cumulative erasing count from physical storage units satisfying a preset writing condition for writing of data for storage, thereby achieving balanced erasing and writing among different physical storage units so that the service lives of different physical storage units are closer to each other, which lowers the possibility of data loss as a result of premature damage to a single physical storage unit or the possibility of impact on the system stability, and thus solves the problem in current technologies.

In the embodiments of the present application, two or more physical storage units are controlled by a central control function module for write control, namely, the central control function module decides which physical storage unit(s) to write data for storage on. Here, a physical storage unit is an entity that is physically independent from other physical storage units and uses a flash memory-based storage medium. It can be an independent flash memory-based physical storage component (e.g., a Flash chip, a hard drive, and the like), a host machine comprising at least one independent flash memory-based physical storage component (e.g., a disk array cabinet, a personal computer, a server), and the like; the central control function module is implemented by software or by a combination of software and hardware, can run on a physical storage unit, or can run on another host machine that is independent from all physical storage units controlled by the central control function module.

FIG. 1 illustrates an example of an application scenario according to some embodiments of the present application, wherein a host machine is installed with a number of hard drives (the physical storage units in the present application scenario), and a storage control module (the central control function module in the present application scenario) runs on the host machine for performing read/write control on all hard drives, including determination of storing what data on which hard drive.

FIG. 2 illustrates another example of an application scenario according to some embodiments of the present application, wherein one host machine in a cluster formed by a plurality of host machines (the physical storage units in the present application scenario) is a main control node, while other host machines are working nodes. The cluster control module (the central control function module in the present application scenario) of the cluster runs on the main control node to perform allocation and management on data storage for the number of host machines. When a fault occurs to the host machine functioning as the main control node, another host machine can be upgraded to the main control node to continue the operations of the cluster.

The central control function module in the application according to some embodiments of

the present application can run on any device capable of computation and storage, for example, a tablet computer, a PC (Personal Computer), a laptop computer, and a server.

In Embodiment I of the present application, a process of a method for writing data for storage is shown in FIG. 3.

Step 310, obtaining a total cumulative erasing count of all blocks in each physical storage unit.

Block is the smallest unit of storage space allocation in data storage. When a block is allocated, an erasing operation will be performed on the block, so that an application (or thread, process, and the like) obtaining the space of the block can use the block for writing data. Therefore, the count that a block is allocated is a cumulative count that the block is erased.

For a physically independent storage component (e.g., a hard drive, a flash chip, and the like), block division is typically performed when the storage component is initialized. With a hard drive as an example, in most file systems, an administrator can designate the size of a block when initializing the hard drive. Assuming that a storage control module of the host machine where the hard drive is located performs storage management on the hard drive, the storage control module can use a unique block identifier among all blocks on the hard drive to represent each block on the hard drive, and maintain a correspondence relationship between each block identifier of the hard drive and a cumulative erasing count. The cumulative erasing count of the block identifier of a block is incremented by 1 each time when the block is allocated; when the hard drive functions as the physical storage unit in the application scenario, the cumulative erasing counts of all the blocks can be added to obtain the total cumulative count of the hard drive. Alternatively, the storage control module can also maintain the total cumulative erasing count of all the blocks in a superblock of the hard drive. The total cumulative erasing count is incremented by the number of blocks allocated each time when blocks are allocated; in other words, the total cumulative erasing count is incremented by 1 each time when a block is allocated. In the manner described above, the total cumulative erasing count of all blocks of a physically independent storage component can be obtained.

In the case where the physical storage unit is a host machine comprising at least one independent physical storage component, the storage control module on each host machine can count the total cumulative erasing count of the physical storage component controlled by the storage control module and report the total cumulative erasing count to a central control function module. The exemplary report manner can be determined according to the implementation of an application scenario. For example, assuming that all host machines form a cluster, a cluster control module of the cluster is the central control function module in the application scenario. Each host machine can proactively report the total cumulative erasing count of all blocks on the

host machine to the cluster control module at a preset period; alternatively, the cluster control module can poll all host machines at a preset period for the total cumulative erasing count of all blocks, and then the cluster control module can receive the total cumulative erasing count of all blocks on each host machine at the preset period.

Step 320, writing data for storage into at least one physical storage unit having the lowest total cumulative erasing count among physical storage units satisfying a preset writing condition.

Assuming that the data for storage needs to be written into N (N being a natural number) physical storage units, one or more conditions are used in current technologies to determine in which physical storage unit(s) to write the data for storage when there are two or more physical storage units that can be used for writing the data for storage. For example, the conditions to be satisfied typically include that a remaining storage space is sufficient to accommodate the data for storage to be written. In some embodiments of the present application, a preset writing condition can be generated according to acceptable ranges of one or more of the above condition(s), which will be described below with two examples:

In the first example, among a plurality of hard drives, N hard drives having the biggest remaining storage space are used as hard drives for writing data for storage. Then in some embodiments of the present application, a preset writing condition can be that the remaining storage space is not smaller than a preset value or not lower than a preset percent of the hard drive capacity.

In the second example, among a number of host machines, N host machines having a remaining storage space that is sufficient to accommodate data for storage to be written and having the lowest number of accesses to the data already stored on the host machines are selected. Then in some embodiments of the present application, preset writing conditions can be that the remaining storage space is sufficient to accommodate the data for storage to be written and that the number of accesses to the data already stored on a host machine does not exceed a threshold.

The data for storage to be written into a physical storage unit can be data for storage to be written when saving a file or can be data for storage to be written when migrating a file. For a central control function module, data for storage to be written when saving a file is typically newly added data for storage, while data for storage to be written when migrating a file means maintenance of data for storage that has already been saved. According to an exemplary implementation of an application scenario, there may be one copy of data for storage to be written when saving a file, or may be two or more copies of data for storage to be written when saving a file, in which case it is typically necessary to write each copy into a different physical storage unit. File migration includes two cases. In one case, a physical storage unit that stores a

file is replaced, i.e., the stored file is written into a new physical storage unit and deleted from the physical storage unit that originally stores the file; in another case where the application scenario stores more than two copies of files, when one of the files is damaged or the physical storage unit on which one of the files is located breaks down, it is necessary to make a copy of the file to a physical storage unit that functions normally to meet the requirement of high availability of the entire system. During file migration, a preset writing condition can be used to exclude physical storage units that have already stored a copy of the file and/or physical storage units that break down.

Therefore, it is possible that more than one physical storage unit is needed for the writing of data for storage for either file storage or file migration. After obtaining a total cumulative erasing count of all blocks for each physical storage unit, therefore, the central control function module determines, according to a preset writing condition, a number of physical storage units available for selection when there is a need for a writing operation on data for storage, and selects, from the physical storage units, N (N being a natural number) physical storage units having the lowest total cumulative erasing counts for the writing of data for storage. As such, balanced erasing and writing can be achieved among different physical storage units.

In the embodiments of the present application, in addition to performing balanced erasing and writing among the physical storage units when writing data for storage, the central control function module can also proactively perform file migration when there is a significant difference in erasing and writing among the physical storage units, so as to balance the use of the physical storage units. For example, the central control function module can monitor the erasing and writing of each physical storage unit. When a difference between total cumulative erasing counts of two physical storage units exceeds a preset deviation range, the data for storage is migrated from one of the two physical storage units that has a higher total cumulative erasing count to the physical storage unit having a lower total cumulative erasing count.

The preset deviation range can be determined according to factors like the size of a file for storage and requirements for balanced erasing and writing in an exemplary application scenario. In an application scenario, the preset deviation range can be determined based on an average of total cumulative erasing counts of all physical storage units. For example, 20% of the average can be used as the preset deviation range.

Therefore, in Embodiment I of the present application, the central control function module selects, based on the statistical data of total cumulative erasing count of all blocks in each physical storage unit, a physical storage unit having the lowest total cumulative erasing count from physical storage units satisfying a preset writing condition for writing of data for storage, thereby achieving balanced erasing and writing among different physical storage units and

avoiding premature damage to a single physical storage unit due to too many times of erasing and writing on the physical storage unit, so that the service lives of different physical storage units are closer to each other, which can lower the possibility of data loss in a non-high availability scenario, and improve the stability of a system where the physical storage unit is located in a high availability scenario.

It should be noted that, if at least some host machines have more than two hard drives or other independent physical storage components in a cluster consisted of a number of host machines, the embodiments of the present application can be applied on the cluster level and the host machine level, respectively. Namely, balanced erasing and writing among different host machines is achieved on the cluster level, and balanced erasing and writing among different hard drives or other physical storage components on a single host machine is achieved on the host machine level. In addition, the FTL balanced erasing and writing algorithm or a file system layer having balanced erasing and writing functions in current technologies can be used on a single host machine or other physical storage components to achieve balanced erasing and writing on the level of physical storage components, thereby achieving balanced erasing and writing on all levels of the entire cluster system, extending the service life of storage devices in the cluster system, and also improving the stability of the cluster system.

In Embodiment II of the present application, a flow chart of a method for writing data for storage in a cluster is shown in FIG. 4. The cluster in this embodiment comprises at least two host machines, and the host machines are subject to storage management by a cluster control module; each host machine comprises at least two hard drives using flash memory as the storage medium, and a storage control module on each host machine performs storage management on the hard drives of the host machine. Embodiment II provides an exemplary implementation of simultaneous application of Embodiment I on two levels (the cluster level and the host machine level), and the description in Embodiment I can be referred to for a detailed description of all steps, which will not be repeated.

Step 410, a cluster control module obtains a total cumulative erasing count of all blocks on each host machine.

Step 420, a storage control module of each host machine obtains a total cumulative erasing count of each hard drive on the host machine, respectively.

Step 430, the cluster controller selects at least one host machine having the lowest host-machine total cumulative erasing count from host machines satisfying a preset host-machine writing condition as a target host machine.

When the cluster has data for storage to be written, for example, data for storage to be written during file migration and/or file storage, assuming that the data for storage needs to be

written to N host machines, the cluster controller selects at least N host machines having the lowest host-machine total cumulative erasing count from host machines satisfying a preset host-machine writing condition as target host machines, i.e., host machines to be written with the data for storage needs to be written.

Step 440, on each target host machine, the storage controller of the host machine writes the data for storage to at least one hard drive that satisfies a preset hard-drive writing condition and has the lowest hard-drive total cumulative erasing count.

When a host machine is selected as a target host machine, assuming that the data for storage needs to be written to M (M being a natural number) hard drives on the host machine, the storage controller of the host machine selects M hard drives having the lowest hard-drive total cumulative erasing count from hard drives satisfying a preset hard-drive writing condition as target hard drives, and writes the data for storage into the target hard drives.

The cluster controller can monitor differences between total cumulative erasing counts of the host machines. When the difference between total cumulative erasing counts of two host machines exceeds a preset host-machine deviation range, the cluster controller migrates the data for storage from one of the two host machines that has a higher total cumulative erasing count to the host machine having a lower total cumulative erasing count. With regard to the host machine to be written with the migrated data for storage, the storage controller of the host machine can write the data to a hard drive that satisfies a preset hard-drive writing condition and has the lowest hard-drive total cumulative erasing count.

The storage controller of each host machine can monitor differences between total cumulative erasing counts of hard drives on the host machine. When the difference between total cumulative erasing counts of two hard drives on a host machine exceeds a preset hard-drive deviation range, the storage controller of the host machine can migrate the data for storage from one of the two hard drives that has a higher total cumulative erasing count to the hard drive having a lower total cumulative erasing count.

Therefore, by applying the method in Embodiment I on the cluster level (between the host machines) and the host machine level (between hard drives on each host machine), respectively, Embodiment II of the present application achieves balanced erasing and writing for all physical storage components in the entire cluster system, extends the service life of storage devices in the cluster system, and also improves the stability of the cluster system.

In an application example of the present application, a high availability cluster system comprises K (K being a natural number greater than 3) host machines, each host machine comprising at least 3 hard drives using flash memory as the storage medium. One copy of each file stored in the cluster should be stored on 3 different host machines, respectively. A cluster

controller of the cluster runs on a main control node (one of the host machines) of the cluster and is used to control which 3 host machines to store a file; a storage controller runs on each host machine and is used to control which hard drive to store the file.

On each host machine, the storage controller maintains, in a superblock of each hard drive, a cumulative erasing count `block_wear_count` for each block of the hard drive and a total cumulative erasing count `disk_wear_count` for all blocks of the hard drive. When a block on the hard drive is allocated, the cumulative erasing count `block_wear_count` corresponding to the block identifier `block_num` of the block is incremented by 1, and the total cumulative erasing count `disk_wear_count` for all blocks of the hard drive is incremented by 1.

In addition, the storage controller maintains a total cumulative erasing count `server_wear_count` for all blocks of the host machine, and `server_wear_count` equals to the sum of `disk_wear_count` of all hard drives on the host machine. Each working node in the cluster must periodically send a heartbeat signal to the main control node, and `server_wear_count` of the host machine where a working node is located can be reported to the cluster controller in the periodic heartbeat signal. The storage controller on the main control node can also report `server_wear_count` of the host machine where the main control node is located to the cluster controller at a period that is the same as that of heartbeat signals.

When a new file is written into the cluster, the cluster controller determines a number of host machines satisfying a preset host-machine writing condition from the K host machines. The preset host-machine writing condition for writing a new file in the present application example is that the remaining storage capacity exceeds 15% of the total storage capacity of a host machine and the number of accesses to the data for storage does not exceed a set threshold. Among the host machines satisfying the preset host-machine writing condition for writing a new file, the cluster controller selects 3 host machines having the lowest values of `server_wear_count` as host machines for writing a new file.

On the selected 3 host machines, a storage controller of each host machine determines a number of hard drives satisfying a preset hard-drive writing condition from the hard drives of the host machine. The preset hard-drive writing condition in the present application example is that the remaining storage capacity exceeds 10% of the total storage capacity of a hard drive. Among the hard drives satisfying the hard-drive writing condition, the storage controller selects a hard drive having the lowest `disk_wear_count` for writing a new file.

When a host machine or a hard drive on a host machine in the cluster breaks down, the cluster controller regards files stored on the host machine or the hard drive no longer useable. As a result, the stored copies of these files in the cluster are fewer than 3 copies. Assuming that a preset host-machine writing condition for migrating a file is that the remaining storage capacity

exceeds 15% of the total storage capacity of a host machine, the number of accesses to the data for storage does not exceed a set threshold, and the file to be written has not been stored, then the cluster controller selects, from the host machines satisfying the preset host-machine writing condition for migrating a file, host machines having the lowest server_wear_count as host machines for writing the migrated files. On the selected host machines, the storage controller selects, from the hard drives satisfying the hard-drive writing condition, a hard drive having the lowest disk_wear_count for writing the migrated file.

The cluster controller monitors a difference between total cumulative erasing counts server_wear_count of the host machines at a certain host machine monitoring period. If the difference between the maximum value and the minimum value of server_wear_count in the cluster exceeds 20% of an average of all server_wear_count, files on the host machine having the maximal server_wear_count are migrated to the host machine having the minimal server_wear_count, until the difference between the maximum value and the minimum value of server_wear_count in the cluster is within 20% of the average of all server_wear_count. Selection of migration files can be implemented with reference to current technologies, which will not be repeated.

On each host machine, the storage controller monitors a difference between total cumulative erasing counts disk_wear_count of the hard drives on the host machine at a certain hard drive monitoring period. If the difference between the maximum value and the minimum value of disk_wear_count on the host machine exceeds 15% of an average of all disk_wear_count, files on the hard drive having the maximal disk_wear_count are migrated to the hard drive having the minimal disk_wear_count, until the difference between the maximum value and the minimum value of disk_wear_count on the host machine is within 15% of the average of all disk_wear_count. Selection of migration files can similarly be implemented with reference to current technologies.

Corresponding to the processes above, embodiments of the present application further provide a device for writing data for storage. The device can be implemented by software, hardware, or a combination of software and hardware. With software implementation as an example, the device in the sense of logic is formed by a CPU (Central Processing Unit) of an apparatus where the device is at reading corresponding computer program instructions into a memory for execution. From a hardware perspective, in addition to the CPU, memory and non-volatile memory shown in FIG. 5, the apparatus where the device for writing data for storage is at typically further comprises other hardware like a chip for wireless signal receiving and transmission and/or other hardware like a board for achieving network communication functions.

FIG. 6 illustrates a device for writing data for storage according to some embodiments of the present application, the device being applied to a central control function module for performing writing control over at least two physical storage units, and the device comprising a total-cumulative-erasing-count member and a physical-storage-unit member, wherein the total-cumulative-erasing-count member is configured to obtain a total cumulative erasing count of all blocks in each physical storage unit; and the physical-storage-unit member is configured to write data for storage into at least one physical storage unit having the lowest total cumulative erasing count among physical storage units satisfying a preset writing condition.

In some examples, the device further comprises a deviation-migration member configured to migrate, when a difference between total cumulative erasing counts of two physical storage units exceeds a preset deviation range, the data for storage from one of the two physical storage units that has a higher total cumulative erasing count to the physical storage unit having a lower total cumulative erasing count.

In the examples, the preset deviation range can be determined based on an average of total cumulative erasing counts of all physical storage units.

Optionally, the data for storage comprises: data for storage to be written when migrating a file and/or data for storage to be written when saving a file.

Optionally, the physical storage unit comprises a host machine; the central control function module comprises a cluster control module of a cluster comprising all host machines; and the total-cumulative-erasing-count member is configured to receive a total cumulative erasing count reported by each host machine at a preset period.

Optionally, the physical storage unit comprises a hard drive; the central control function module comprises a storage control module of a host machine; and the total-cumulative-erasing-count member is configured to read a total cumulative erasing count maintained in a superblock of each hard drive, the total cumulative erasing count being incremented by 1 after each block of the hard drive is allocated.

FIG. 7 illustrates a cluster according to some embodiments of the present application. The cluster comprises at least two host machines, and each host machine comprises at least two hard drives using flash memory as storage medium. The cluster further comprises a cluster control module and a storage control module on each host machine, wherein the cluster control module is configured to obtain a total cumulative erasing count of all blocks on each host machine, and when writing data for storage is needed, select at least one host machine having the lowest host-machine total cumulative erasing count from host machines satisfying a preset host-machine writing condition as a target host machine; the storage control module on each host machine is configured to obtain a total cumulative erasing count of each hard drive on the host machine, and

when the host machine is selected as a target host machine, write the data for storage to at least one hard drive that satisfies a preset hard-drive writing condition and has the lowest hard-drive total cumulative erasing count.

Optionally, the cluster control module is further configured to migrate, when a difference between total cumulative erasing counts of two host machines exceeds a preset host-machine deviation range, the data for storage from one of the two host machines that has a higher total cumulative erasing count to the host machine having a lower total cumulative erasing count; the storage control module on each host machine is further configured to migrate, when a difference between total cumulative erasing counts of two hard drives on a host machine exceeds a preset hard-drive deviation range, the data for storage from one of the two hard drives that has a higher total cumulative erasing count to the hard drive having a lower total cumulative erasing count

Optionally, the data for storage comprises: data for storage to be written when migrating a file.

Only preferred embodiments of the present application are described above, which are not used to limit the present application. Any modification, equivalent substitution or improvement made within the spirit and principle of the present application shall be encompassed by the protection scope of the present application.

In a typical configuration, a computation device includes one or more processors (CPUs), input/output interfaces, network interfaces, and a memory.

The memory may include computer readable media, such as a volatile memory, a Random Access Memory (RAM), and/or a non-volatile memory, e.g., a Read-Only Memory (ROM) or a flash RAM. The memory is an example of a computer readable medium.

Computer readable media include permanent, volatile, mobile and immobile media, which can implement information storage through any method or technology. The information may be computer readable instructions, data structures, program modules or other data. Examples of storage media of computers include, but are not limited to, Phase-change RAMs (PRAMs), Static RAMs (SRAMs), Dynamic RAMs (DRAMs), other types of Random Access Memories (RAMs), Read-Only Memories (ROMs), Electrically Erasable Programmable Read-Only Memories (EEPROMs), flash memories or other memory technologies, Compact Disk Read-Only Memories (CD-ROMs), Digital Versatile Discs (DVDs) or other optical memories, cassettes, cassette and disk memories or other magnetic memory devices or any other non-transmission media, which can be used for storing information accessible to a computation device. According to the definitions herein, the computer readable media do not include transitory media, such as modulated data signals and carriers.

It should be further noted that the terms of “including”, “comprising” or any other variants

thereof intend to encompass a non-exclusive inclusion, so that a process, method, commodity or device comprising a series of elements not only comprises these elements, but also comprises other elements that are not specifically listed, or further comprises elements that are inherent to the process, method, commodity or device. When there is no further restriction, elements defined by the statement “comprising one...” do not exclude that a process, method, commodity or device comprising the above elements further comprises additional identical elements.

A person skilled in the art should understand that the embodiments of the present application may be provided as a method, a system, or a computer program product. Therefore, the present application may be implemented as a complete hardware embodiment, a complete software embodiment, or an embodiment combining software and hardware. Moreover, the present application may be in the form of a computer program product implemented on one or more computer usable storage media (including, but not limited to, a magnetic disk memory, CD-ROM, an optical memory, and the like) comprising computer usable program codes therein.

Claims

1. A method for writing data in a cluster of a plurality of host machines, each host machine comprising a plurality of hard drives, the method comprising:

obtaining a plurality of erasing counts for the hard drives on each host machine of the host machines, wherein the erasing count for each hard drive of the hard drives equals a total count of erasing operations that have been performed on the hard drive;

obtaining a total erasing count for each host machine of the host machines, wherein the total erasing count equals a sum of the erasing counts of the hard drives on the host machine;

obtaining data to be written into the cluster;

in response to obtaining the data to be written into the cluster, identifying one or more of the host machines that each satisfy first conditions associated with writing the data, wherein the first conditions comprise:

that a remaining storage capacity of a host machine exceeds a threshold percentage of a total storage capacity of the host machine,

that a number of accesses to data already stored on a host machine does not exceed an access threshold, and

that the total erasing count of a host machine does not exceed a threshold value;

determining a host machine that has a smallest total erasing count among the identified one or more host machines;

identifying one or more hard drives on the determined host machine that each satisfy one or more second conditions associated with writing the data;

determining a hard drive that has the smallest erasing count among the identified one or more hard drives; and

writing the data in the determined hard drive.

2. The method according to claim 1, further comprising: when a difference between the erasing counts of two hard drives of a host machine exceeds a preset deviation range, migrating data from one of the two hard drives that has a higher erasing count to the hard drive that has a lower erasing count.

3. The method according to claim 2, wherein the preset deviation range is determined based on an average of the erasing counts of all of the hard drives of the host machine.

4. The method according to claim 1, wherein the data to be written into the cluster comprises: data to be written for file migration and/or data to be written for file storage.

5. The method according to any one of the preceding claims 1 to 4, wherein the obtaining a total erasing count for each host machine of the host machines comprises obtaining a total erasing count from each host machine at a preset period.

6. The method according to any one of the preceding claims 1 to 4, wherein the obtaining a plurality of erasing counts for the hard drives on each host machine of the host machines comprises reading an erasing count maintained in a superblock of each hard drive, the erasing count being incremented by 1 after each block of the hard drive is allocated.

7. A device for writing data for storage in a cluster of a plurality of host machines, each host machine comprising a plurality of hard drives, the device comprising: a processor and computer-readable storage medium configured with instructions executable by the processor to cause the processor to perform operations comprising:

- obtaining a plurality of erasing counts for the hard drives on each host machine of the host machines, wherein the erasing count for each hard drive of the hard drives equals a total count of erasing operations that have been performed on the hard drive;

- obtaining a total erasing count for each host machine of the host machines, wherein the total erasing count equals a sum of the erasing counts of the hard drives on the host machine;

- obtaining data to be written into the cluster;

- in response to obtaining the data to be written into the cluster, identifying one or more of the host machines that each satisfy first conditions associated with writing the data, wherein the first conditions comprise:

 - that a remaining storage capacity of a host machine exceeds a threshold percentage of a total storage capacity of the host machine,

 - that a number of accesses to data already stored on a host machine does not exceed an access threshold, and

 - that the total erasing count of a host machine does not exceed a threshold value;

- determining a host machine that has a smallest total erasing count among the identified one or more host machines;

- identifying one or more hard drives on the determined host machine that each satisfy one or more second conditions associated with writing the data;

- determining a hard drive that has the smallest erasing count among the identified one or more hard drives; and

- writing the data in the determined hard drive.

8. The device according to claim 7, wherein the operations further comprise when a difference between the erasing counts of two hard drives of a host machine exceeds a preset deviation range, migrating data for storage from one of the two hard drives that has a higher erasing count to the hard drive that has a lower erasing count.

9. The device according to claim 8, wherein the preset deviation range is determined based on an average of the erasing counts of all of the hard drives of the host machine.

10. The device according to claim 7, wherein the data to be written into the cluster comprises: data to be written for file migration and/or data to be written for file storage.

11. The device according to any one of the preceding claims 7 to 10, wherein the obtaining a total erasing count for each host machine of the host machines comprises obtaining a total erasing count from each host machine at a preset period.

12. The device according to any one of the preceding claims 7 to 10, wherein the obtaining a plurality of erasing counts for the hard drives on each host machine of the host machines comprises reading an erasing count maintained in a superblock of each hard drive, the erasing count being incremented by 1 after each block of the hard drive is allocated.

13. A non-transitory computer-readable storage medium for writing data in a cluster of a plurality of host machines, each host machine comprising a plurality of hard drives, configured with instructions executable by one or more processors to cause the one or more processors to perform operations comprising:

- obtaining a plurality of erasing counts for the hard drives on each host machine of the host machines, wherein the erasing count for each hard drive of the hard drives equals a total count of erasing operations that have been performed on the hard drive;

- obtaining a total erasing count for each host machine of the host machines, wherein the total erasing count equals a sum of the erasing counts of the hard drives on the host machine;

- obtaining data to be written into the cluster;

- in response to obtaining the data to be written into the cluster, identifying one or more of the host machines that each satisfy first conditions associated with writing the data, wherein the first conditions comprise:

 - that a remaining storage capacity of a host machine exceeds a threshold percentage of a total storage capacity of the host machine,

 - that a number of accesses to data already stored on a host machine does not exceed an access threshold, and

 - that the total erasing count of a host machine does not exceed a threshold value;

 - determining a host machine that has a smallest total erasing count among the identified one or more host machines;

 - identifying one or more hard drives on the determined host machine that each satisfy one or more second conditions associated with writing the data;

 - determining a hard drive that has the smallest erasing count among the identified one or more hard drives; and

 - writing the data in the determined hard drive.

14. The non-transitory computer-readable storage medium according to claim 13, wherein

the operations further comprise:

when a difference between the erasing counts of two hard drives of a host machine exceeds a preset deviation range, migrating data for storage from one of the two hard drives that has a higher erasing count to the hard drive that has a lower erasing count.

15. The non-transitory computer-readable storage medium according to claim 14, wherein the preset deviation range is determined based on an average of the erasing counts of all of the hard drives of the host machine.

16. The non-transitory computer-readable storage medium according to claim 13, wherein the data to be written into the cluster comprises: data to be written for file migration and/or data to be written for file storage.

17. The non-transitory computer-readable storage medium according to any one of the preceding claims 13 to 16, wherein

the obtaining a total erasing count for each host machine of the host machines comprises obtaining a total erasing count from each host machine at a preset period.

18. The non-transitory computer-readable storage medium according to any one of the preceding claims 13 to 16, wherein

the obtaining a plurality of erasing counts for the hard drives on each host machine of the host machines comprises reading an erasing count maintained in a superblock of each hard drive, the erasing count being incremented by 1 after each block of the hard drive is allocated.