



(12) 发明专利

(10) 授权公告号 CN 108628542 B

(45) 授权公告日 2021.08.03

(21) 申请号 201710175180.0

(22) 申请日 2017.03.22

(65) 同一申请的已公布的文献号  
申请公布号 CN 108628542 A

(43) 申请公布日 2018.10.09

(73) 专利权人 华为技术有限公司  
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72) 发明人 任波 黄飞腾

(74) 专利代理机构 广州三环专利商标代理有限公司 44202  
代理人 熊永强 李稷芳

(51) Int. Cl.  
G06F 3/06 (2006.01)  
G06F 16/16 (2019.01)

(56) 对比文件

- CN 106156070 A, 2016.11.23
- WO 2017020576 A1, 2017.02.09
- CN 103744628 A, 2014.04.23
- CN 103186350 A, 2013.07.03
- CN 103593436 A, 2014.02.19
- CN 103812877 A, 2014.05.21
- CN 105447059 A, 2016.03.30

梅飞. “SSDKV:一种SSD友好的键值对存储系统”.《计算机工程与科学》.2016,第38卷(第7期),

审查员 王永贵

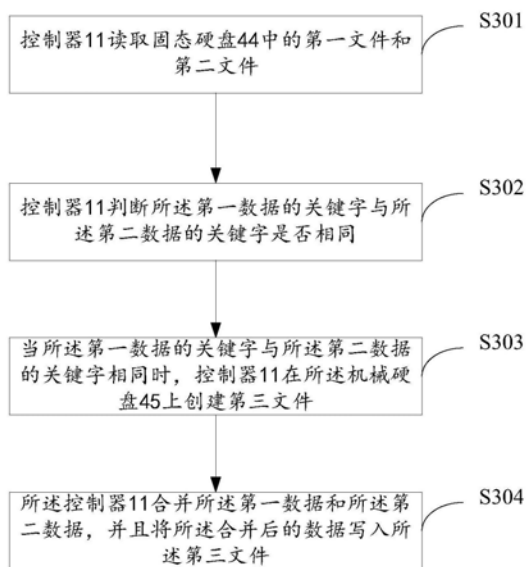
权利要求书2页 说明书7页 附图3页

(54) 发明名称

一种文件合并方法及控制器

(57) 摘要

本发明实施例提供了一种文件合并方法。该方法应用于存储系统中。所述存储系统包括控制器、固态硬盘和机械硬盘。所述控制器读取所述固态硬盘中的第一文件和第二文件。第一文件中保存有第一数据以及所述第一数据的关键字，第二文件中保存有第二数据以及所述第二数据的关键字。所述控制器判断所述第一数据的关键字与所述第二数据的关键字是否相同，当所述第一数据的关键字与所述第二数据的关键字相同时，所述控制器在所述机械硬盘上创建第三文件。然后，所述控制器合并所述第一数据和所述第二数据，并将所述合并后的数据写入所述第三文件。可以提高读文件的效率。



1. 一种文件合并方法,其特征在于,所述方法应用于存储系统中,所述存储系统包括控制器、固态硬盘和机械硬盘,所述方法包括:

所述控制器读取所述固态硬盘中的第一文件和第二文件,其中,所述第一文件中保存有第一数据以及所述第一数据的关键字,所述第二文件中保存有第二数据以及所述第二数据的关键字;

判断所述第一数据的关键字与所述第二数据的关键字是否相同;当所述第一数据的关键字与所述第二数据的关键字相同时,所述控制器在所述机械硬盘上创建第三文件;所述控制器合并所述第一数据和所述第二数据,并将所述合并后的数据写入所述第三文件;

其中,

所述存储系统还包括储存级存储器,所述储存级存储器中还保存有日志文件,所述方法还包括:

所述控制器回放所述日志文件,得到被写入到所述日志文件中的所述第一数据以及所述第一数据的关键字以及所述第二数据以及所述第二数据的关键字;

所述控制器在所述固态硬盘中创建所述第一文件,将所述第一数据以及所述第一数据的关键字写入所述第一文件;所述控制器在所述固态硬盘中创建所述第二文件,将所述第二数据以及所述第二数据的关键字写入所述第二文件。

2. 根据权利要求1所述的方法,其特征在于,所述控制器的缓存中保存有至少两个不可变跳表和一个可变跳表,所述第一数据来自于所述至少两个不可变跳表中一个不可变跳表中的数据,所述第二数据来自于所述至少两个不可变跳表中另一个不可变跳表中的数据,所述可变跳表用于接收主机发送的数据。

3. 根据权利要求1所述的方法,其特征在于,还包括:所述控制器监控所述固态硬盘中的文件的数量是否达到预设阈值,当所述固态硬盘中的文件的数量已达到所述预设阈值时,所述控制器读取所述固态硬盘中的所述第一文件和所述第二文件。

4. 根据权利要求1所述的方法,其特征在于,所述回放所述日志文件之前还包括:

所述控制器接收主机发送的第一写数据指令,将所述第一写数据指令写入所述日志文件中,所述第一写数据指令包括所述第一数据以及所述第一数据的关键字;

所述控制器接收主机发送的第二写数据指令,将所述第二写数据指令写入所述日志文件中,所述第二写数据指令包括所述第二数据以及所述第二数据的关键字。

5. 根据权利要求4所述的方法,其特征在于,所述将所述第一写数据指令写入所述日志文件中包括采用追加写的方式将所述第一写数据指令写入所述日志文件。

6. 根据权利要求4或5所述的方法,其特征在于,所述日志文件还包括其他写数据指令,所述方法还包括:所述控制器对所述日志文件中的所有写数据指令按照每个写数据指令中携带的关键字进行排序。

7. 一种控制器,其特征在于,所述控制器位于存储系统中,所述存储系统包括所述控制器、固态硬盘和机械硬盘,所述控制器包括:

通信接口,用于与所述固态硬盘通信,以及用于与所述机械硬盘通信;

处理器,用于所述固态硬盘中的第一文件和第二文件,其中,所述第一文件中保存有第一数据以及所述第一数据的关键字,所述第二文件中保存有第二数据以及所述第二数据的关键字;判断所述第一数据的关键字与所述第二数据的关键字是否相同;当所述第一数据

的关键字与所述第二数据的关键字相同时,在所述机械硬盘上创建第三文件;合并所述第一数据和所述第二数据,并将所述合并后的数据写入所述第三文件;

所述存储系统还包括储存级存储器,所述储存级存储器中还保存有日志文件,

其中,所述处理器还用于回放所述日志文件,得到被写入到所述日志文件中的所述第一数据以及所述第一数据的关键字以及所述第二数据以及所述第二数据的关键字;在所述固态硬盘中创建所述第一文件,将所述第一数据以及所述第一数据的关键字写入所述第一文件;在所述固态硬盘中创建所述第二文件,将所述第二数据以及所述第二数据的关键字写入所述第二文件。

8. 根据权利要求7所述的控制器,其特征在于,所述控制器还包括缓存,所述缓存中保存有至少两个不可变跳表和一个可变跳表,所述第一数据来自于所述至少两个不可变跳表中一个不可变跳表中的数据,所述第二数据来自于所述至少两个不可变跳表中另一个不可变跳表中的数据,所述可变跳表用于接收主机发送的数据。

9. 根据权利要求7所述的控制器,其特征在于,所述处理器还用于监控所述固态硬盘中的文件的数量是否达到预设阈值,当所述固态硬盘中的文件的数量已达到所述预设阈值时,所述控制器读取所述固态硬盘中的所述第一文件和所述第二文件。

10. 根据权利要求7所述的控制器,其特征在于,所述处理器,还用于回放所述日志文件之前,接收主机发送的第一写数据指令,将所述第一写数据指令写入所述日志文件中,所述第一写数据指令包括所述第一数据以及所述第一数据的关键字;

接收主机发送的第二写数据指令,将所述第二写数据指令写入所述日志文件中,所述第二写数据指令包括所述第二数据以及所述第二数据的关键字。

11. 根据权利要求10所述的控制器,其特征在于,所述将所述第一写数据指令写入所述日志文件中包括采用追加写的方式将所述第一写数据指令写入所述日志文件。

12. 根据权利要求10或11所述的控制器,其特征在于,所述日志文件还包括其他写数据指令,所述处理器还用于,对所述日志文件中的所有写数据指令按照每个写数据指令中携带的关键字进行排序。

13. 一种计算机可读取存储介质,其特征在于,

所述计算机可读取存储介质存储有计算机程序,所述计算机程序被处理器执行时能够实现权利要求1至6任意一项所述的方法。

## 一种文件合并方法及控制器

### 技术领域

[0001] 本发明实施例涉及存储技术领域,尤其涉及一种文件合并方法及控制器。

### 背景技术

[0002] 日志结构合并树(log structured merge tree,LSMtree)是相对应B+树的一种数据结构,其主要是为了克服B+树的大量随机IO导致性能低的问题,本质是实现读写之前取得平衡,牺牲读性能来大幅度提高写性能。

[0003] 业界有很多基于LSMtree原理实现的KV存储引擎,这些存储引擎被广泛应用各种数据存储系统中。这些应用实现KV存储引擎的核心思想都是将随机IO转化为顺序IO写入,数据再通过内存整理为局部有序的文件存储到文件系统,最后再对多个文件系统中的文件进行合并和排序提升读取性能。然而,如何提高这些文件的读数据的效率是当前需要解决的问题。

### 发明内容

[0004] 本发明第一方面提供了一种文件合并方法。该方法应用于存储系统中。所述存储系统包括控制器、固态硬盘和机械硬盘。所述控制器读取所述固态硬盘中的第一文件和第二文件。第一文件中保存有第一数据以及所述第一数据的关键字,第二文件中保存有第二数据以及所述第二数据的关键字。所述控制器判断所述第一数据的关键字与所述第二数据的关键字是否相同,当所述第一数据的关键字与所述第二数据的关键字相同时,所述控制器在所述机械硬盘上创建第三文件。然后,所述控制器合并所述第一数据和所述第二数据,并将所述合并后的数据写入所述第三文件。

[0005] 按照本发明第一方面提供的文件合并方法,将合并前的文件存储在固态硬盘中,将合并后的文件存储在机械硬盘中。由于合并前的文件中的数据属于新数据,被读取的可能性较大,将其存储在固态硬盘中可以提高读数据的效率。而合并后的文件中的数据属于旧数据,被读取的可能性较小,将其存储在机械硬盘中,可以降低成本。

[0006] 在第一方面中,也可以将合并后的文件,即第三文件保存在固态硬盘中,虽然不能达到降低成本的目的,但可以进一步提高读数据的效率。

[0007] 基于第一方面,在第一方面的第一种实现方式中,所述控制器的缓存中保存有至少两个不可变跳表和一个可变跳表,所述第一数据来自于所述至少两个不可变跳表中一个不可变跳表中的数据,所述第二数据来自于所述至少两个不可变跳表中另一个不可变跳表中的数据,所述可变跳表用于接收主机发送的数据。

[0008] 基于第一方面,在第一方面的第二种实现方式中,所述控制器监控所述固态硬盘中的文件的数量是否达到预设阈值,当所述固态硬盘中的文件的数量已达到所述预设阈值时,所述控制器读取所述固态硬盘中的所述第一文件和所述第二文件。这里是给文件合并设置了一个触发条件,即,当固态硬盘中的文件的数量达到预设阈值时,开始执行文件合并。关于如何选择待合并的文件,可以是将所述固态硬盘中的所有文件作为待合并文件,也

可以由用户指定待合并的文件等方式。

[0009] 基于第一方面,在第一方面的第三种实现方式中,所述存储系统还包括储存级存储器,所述储存级存储器中还保存有日志文件。所述控制器接收主机发送的第一写数据指令,将所述第一写数据指令写入所述日志文件,所述第一写数据指令包括所述第一数据以及所述第一数据的关键字。所述控制器接收主机发送的第二写数据指令,将所述第二写数据指令写入所述日志文件中,所述第二写数据指令包括所述第二数据以及所述第二数据的关键字。所述控制器回放所述日志文件,得到所述第一数据以及所述第一数据的关键字以及所述第二数据以及所述第二数据的关键字。所述控制器在所述固态硬盘中创建所述第一文件,将所述第一数据以及所述第一数据的关键字写入所述第一文件,在所述固态硬盘中创建所述第二文件,将所述第二数据以及所述第二数据的关键字写入所述第二文件。由于储存级存储器是一种读写效率较高的新型存储介质,将日志文件保存在储存级存储器中,通过所述日志文件回放得到数据的效率会更高

[0010] 结合第一方面的第三种实现方式,在第一方面的第四种实现方式中将所述第一写数据指令写入所述日志文件中具体是采用追加写的方式。采用追加写的方式可以提高写日志文件的效率。

[0011] 结合第一方面的第三种或第四种实现方式,在第一方面的第五种实现方式中,所述日志文件还包括其他写数据指令,所述方法还包括:所述控制器对所述日志文件中的所有写数据指令按照每个写数据指令中携带的关键字进行排序。经过排序后的日志文件,经回放生成文件数据之后,数据的关键字也是按照从小到大或者从大到小的顺序进行排序的,方便查找。

[0012] 本发明第二方面提供了一种控制器。所述控制器位于存储系统中,所述存储系统包括所述控制器、固态硬盘和机械硬盘。所述控制器包括通信接口和处理器。其中,通信接口,用于与所述固态硬盘,以及所述机械硬盘通信。所述处理器用于执行第一方面任意一种实现方式的方法。

## 附图说明

[0013] 为了更清楚地说明本发明实施例的技术方案,下面将对实施例描述中所需要使用的附图作简单地介绍。

[0014] 图1为本发明实施例提供的一种存储系统的组成图;

[0015] 图2为本发明实施例提供的一种控制器11的结构示例图;

[0016] 图3为本发明实施例提供的一种控制器11组织数据的结构图;

[0017] 图4为本发明实施例提供的一种文件合并方法的流程图。

## 具体实施方式

[0018] 下面将结合附图,对本发明实施例中的技术方案进行清楚、完整地描述。

[0019] KV(key-value)存储系统的数据存储采用LSM-Tree的结构实现。LSM-Tree将硬盘的随机写转化为顺序写,从而提高了写数据的速度。本发明实施例将LSM-Tree应用于包含固态硬盘的存储系统中。下面结合图1和图2,对本实施例所描述的存储系统进行介绍。

[0020] 如图1所示,存储系统包括至少一个控制器11、至少一个固态硬盘44和至少一个机

械硬盘45。

[0021] 控制器11通过存储区域网络(英文:storage area network,SAN)与主机(图中未示出)连接。控制器11可以是一种计算设备,如服务器、台式计算机等等。在控制器11上安装有写入系统以及应用程序。控制器11可以接收来自主机的输入输出(I/O)请求。控制器11还可以存储I/O请求中携带的数据,并且将该数据写入固态硬盘44中。控制器11属于系统控制器,系统控制器是独立的设备,不同于固态硬盘44中的控制芯片。本实施例将固态硬盘44的控制芯片称为固态硬盘控制器。如无特别指明,本实施例中的控制器都是指系统控制器。

[0022] 固态硬盘44(英文:Solid State Disk,SSD)是以闪存(英文:flash memory)芯片为存储介质的存储器,又名固态驱动器(Solid State Drive,SSD)。

[0023] 图2是控制器11的结构示例图,如图2所示,控制器11包括接口卡110、处理器112、存储器111和接口卡113。

[0024] 接口卡110,用于和主机通信,控制器11可以通过接口卡110接收主机的写数据指令。例如,写数据指令包括关键字(英文:key)和值(英文:value),所述值是指待写入所述存储系统的数据(以下简称为数据),所述关键字是所述值的标识。举一个具体的例子,所述值可以是学生的各种信息,所述关键字可以是该学生的学号,或者其他表示该学生某一方面属性的标识。

[0025] 接口卡113,用于和固态硬盘44通信,控制器11可以通过接口卡113将写数据请求(包括上面描述的写数据指令中的关键字和数据,以及控制器11为所述数据分配的逻辑地址)发送给固态硬盘44存储。

[0026] 处理器112,可能是一个中央处理器(英文:centeral processing unit,CPU)。在本发明实施例中,处理器112可以用于接收来自主机的写数据指令或者读取指令并且处理这些指令。处理器112还可以将所述写数据指令中的数据发送给固态硬盘44。此外,处理器112,还可以用于在一定条件时,执行文件的合并操作。存储器111,1包括易失性存储器,非易失性存储器或其组合。易失性存储器例如为随机访问存储器(英文:random-access memory,RAM)。非易失性存储器例如软盘、硬盘、固态硬盘44(solid state disk,SSD)、光盘等各种可以存储程序代码的机器可读介质。存储器111具有保电功能,保电功能是指系统发生掉电又重新上电时,存储器111中存储的数据也不会丢失。存储器111可以有一个或者多个,用于临时存储从主机接收的数据或从固态硬盘44读取的数据,例如控制器11接收主机发送的多个写数据请求时,可以将所述多个写数据请求中的数据暂时保存在存储器111中。此外,存储器111还保存有可变存储表(英文:memtable)和不可变存储表(immutable memtable)。根据LSM-Tree机制,memtable用于接收一段时间内主机发送的写数据指令。示例性的,memtable中存储的写数据指令可以如表1所示:

[0027]

Key	Value	Version
0001	张三	2017.3
0003	李四	2017.3
.....	.....	.....
0002	王五	2017.3

[0028] 表1

[0029] 如表1所示,memtable中保存有多组数据,每组数据都包括关键字、值和版本信息。

关键字和值已在上面的段落中进行了介绍,这里不再赘述。版本信息是指数据写入的时间信息,或者其他用以标识数据写入的先后顺序的信息。可以理解的是,本实施例中版本信息可以精确到月,也可以精确到日、时、分、秒。本实施例并不对版本信息的形式和精确度进行任何限定。另外,只要在这段时间内写入的数据都会保存在所述memtable中。所述版本信息是控制器11接收写数据指令时为所述写数据指令分配的。

[0030] 根据LSM-Tree机制,当memtable中保存的所有数据的数据量达到预设阈值时,所述memtable将转化为immutable memtable。immutable memtable会不再接收新的数据,同时存储器111中会再创建一个新的memtable。所述新的memtable将继续接收数据。因此存储器111中可能保存有一个或多个immutable memtable,但只有一个memtable。每个immutable memtable中的数据是固化的,而memtable可以不断接收新写入的数据。immutable memtable中的数据需要按照关键字从小到大或者从大到小进行排序,如表2所示:

[0031]

Key	Value	Version
0001	张三	2017.3
0001	张三	2017.4
0002	王五	2017.3
0002	王五	2017.4
.....	.....	.....
0003	李四	2017.3

[0032] 在LSM-Tree机制中,每个immutable memtable中的数据被倾倒(英文:dump)在一个分类线表(Sorted String Table,SST)文件(以下简称为文件)中。可以理解的是,所述存储系统中包含多个文件,每个文件与一个immutable memtable对应,用于接收由所述immutable memtable倾倒而来的数据。本实施例中将由immutable memtable倾倒而来的文件(也就是未经过合并的文件)作为第一层级的文件。

[0033] 本实施例中,控制器11包含文件系统,因此可以以文件的形式组织数据。具体的,控制器11组织数据的结构如图3所示。如图3所示,key-value数据保存在固态硬盘的44的擦除块(英文:block)中,该数据保存在擦除块中的实际地址为该数据的物理地址。控制器11将一个或多个擦除块映射为一个逻辑空间,举例来说,所述逻辑空间可以是逻辑单元(logical unit,LU)。控制器11和固态硬盘44都保存有该数据的逻辑地址,以及所述逻辑地址与物理地址之间的对应关系。该数据的逻辑地址是指该数据所在的逻辑单元的标识(例如,LU ID)、该数据位于该逻辑单元内的起始位置(例如,Logical block Address)和长度。另外,由于控制器11在逻辑空间的上层还具有文件系统,因此用户可以以文件的形式访问数据。控制器11还保存有所述逻辑地址与文件的标识之间的对应关系,以及所述物理地址与文件的标识之间的对应关系。

[0034] 由上面的描述可知,每个immutable memtable中包含不同关键字的数据,那么,相应地,与immutable memtable对应的文件中保存的也是不同关键字的数据。也就是说,同样的关键字所对应的数据会分布在不同的文件中。因此,当接收读取指令要求读取数据时,固态硬盘44需要根据读取指令携带的关键字从不同的文件中获取所述关键字对应的各个数据,并根据版本信息确认出最新的数据返回给用户。这样的过程会影响读取数据的效率,因

此有必要对文件进行合并。合并的目的是对不同文件中,关键字相同的数据进行合并,合并后同一个关键字只对应一个数据或少量的几个数据(根据系统设置或者用户需求设定)。

[0035] 具体的,当第一层级的文件的总量达到一定阈值时,需要将第一层级中的文件进行合并。其合并方式可以是,对于第一层级中的所有文件进行合并,也可以是第一层级中的部分文件进行合并,还可以是由用户指定第一层级中待合并的文件。合并后生成的新的文件,就是第二层级的文件。依此类推,当第二层级的文件的总量达到一定阈值时,可以对其进行再次规定,建立第三层级的文件。对文件进行合并可以减少重复的数据或者无效的数据,提高读数据的效率。例如,在一个时间段内file1中写入了键范围为[1-10]的数据,file4更新了键范围为[5-20]的数据,file7又删除了[10-20]的部分数据。

[0036] 另外,在SST文件中,每个数据的存储格式为键值(Key-Value,KV),通过键值中的关键字标识文件中的每个数据,每个文件保存的所有关键字是有序排列的,通过起始关键字与终止关键字的组合标识每个文件的键范围。

[0037] 一般来说,第一层级的文件中的数据被读取的可能性更大,因为这些数据是新接收的数据,而其他层级的文件中的数据属于旧数据,被读取的可能性相对较小。因此,在本实施例中,将第一层级的文件部署在固态硬盘44中,将除第一层级以外的其他层级的文件部署在机械硬盘45中。由于固态硬盘读写数据的效率高于机械硬盘,因此采用这样的文件部署,可以进一步提高读数据的效率。

[0038] 另外,图1所示的存储系统还可以包含储存级存储器(storageclass memory,SCM)46,用于保存日志文件,控制器11在接收写数据指令之后,首先将写数据指令携带的信息写入所述日志文件,再将所述写数据指令写入memtable。上面介绍了将immutable memtable中的数据倾倒在固态硬盘44中,生成一个SST文件。另外,本实施例还提供了另一种数据存储的方式,即,直接根据所述日志文件中的记录生成SST文件。具体的,控制器11在接收主机发送的多个写数据指令之后,采用追加写的方式将每个写数据指令写入所述日志文件。因此,所述日志文件中保存有多个写数据指令,其中每个写数据指令都携带有关键字。所述日志文件按照每个写数据指令的关键字进行排序。另外,所述日志文件中还包括控制信息,例如操作类型、全局序列号等信息。这些信息用于日志文件回放后生成数据以及数据的关键字,以及保证回放时的顺序和操作的可靠性。操作类型用于指示写数据指令或者删除指令等操作。全局序列号可以是日志序列号(log sequence number,LSN)。

[0039] 另外,存储系统中还保存有一个或多个管理文件(图3中未示出),用于记录各个文件所在的层级,单个文件最大的关键字以及最小的关键字,每个文件的标识,每个文件所包含的关键字的区间以及其他元数据信息。其他元数据信息包括但不限于:数据在固态硬盘44中的实际地址,逻辑地址和对数据进行写入的写入记录,以及实际地址与所属文件的标识的对应关系等,数据的逻辑地址可以是数据所属的文件的标识,也可以是数据所属的文件的标识和数据在文件中的偏移位置的组合,也可以是数据所属的文件的标识,数据在文件中的偏移位置和数据长度的组合。写入记录包括对用户数据进行写入的方式和时间,通过版本信息对数据进行写入的时间。所述管理文件可以保存在固态硬盘44中,也可以保存在机械硬盘45中,还可以保存在储存级存储器46中。

[0040] 下面结合图1所示的存储系统,以及图2所示的控制器11的结构,对本发明实施例公开的文件合并方法进行具体的描述。请参见图4,图4为本发明实施例提供的一种文件合



并方法的流程图。如图4所示,所述数据的存储方法由控制器11中的处理器112执行,具体包括如下步骤。

[0041] S301,控制器11读取固态硬盘44中的第一文件和第二文件。

[0042] 一种情况下,第一文件中的数据和第二文件中的数据均来自控制器11的存储器111。具体的,第一文件是由存储器111中的一个immutable memtable倾倒而来的,第二文件是由存储器111中的另一个immutable memtable倾倒而来的。由于immutable memtable中的数据是按照关键字从小到大或者从大到小的顺序进行排序的,因此第一文件和第二文件中的数据也是按照关键字排序的。这是为了提高读数据的效率。

[0043] 另一种情况下,第一文件中的数据和第二文件中的数据均是由日志文件回放而来的。由于日志文件中保存有多个写数据指令,以及操作类型、全局序列号等信息。因此,可以由日志文件回放生成第一文件的数据和第二文件的数据。其中,每个文件的尺寸是预先设置的,可以和一个immutable memtable的尺寸相同,也可以和一个immutable memtable的尺寸不同。在这种情况下,日志文件通常保存在储存级存储器中,以利用更加快速地从所述日志文件获取数据。

[0044] 无论第一文件的数据和第二文件的数据是来自存储器111还是来自日志文件,当所述第一文件的数据和所述第二文件的数据在固态硬盘44中进行持久化存储之后,可以将日志文件中对应的内容删掉,以节省日志文件所占用的存储空间。

[0045] 以控制器11读取第一文件为例,控制器11根据第一文件的标识在文件的标识与文件所包含的数据的逻辑地址的对应关系中查询,获取第一文件所包含的数据(例如,第一数据)的逻辑地址;控制器11将所述逻辑地址发送给固态硬盘44,固态硬盘44根据所述逻辑地址和物理地址的对应关系,从所述逻辑地址对应的物理地址中获取数据并发送给控制器11。同理,控制器11也按照这样的方式读取第二文件。

[0046] 为了方便描述,本实施例中第一文件包含第一数据以及第一数据的关键字,第二文件包含第二数据以及第二数据的关键字。

[0047] S302,控制器11判断所述第一数据的关键字与所述第二数据的关键字是否相同。

[0048] 数据的关键字是用来标识所述数据的,判断第一数据的关键字与所述第二数据的关键字是否相同是为了确定是否需要所述第一文件和第二文件进行合并。

[0049] S303,当所述第一数据的关键字与所述第二数据的关键字相同时,控制器11在所述机械硬盘45上创建第三文件。

[0050] 如果第一数据的关键字与所述第二数据的关键字相同,则说明第一文件和第二文件需要合并,合并后的文件为第三文件。由于合并后的文件中的数据属于旧数据,被读取的可能性比第一文件、第二文件的可能性小,因此从节省成本的角度考虑,控制器11将合并后的数据写入机械硬盘45。具体的,控制器11创建第三文件,为所述第三文件分配标识,并且为所述第三文件分配一段逻辑地址区间,所述逻辑地址区间与机械硬盘45的一段物理地址区间对应。控制器11保存所述第三文件的标识与所述逻辑地址区间的对应关系,以及所述逻辑地址区间与所述物理地址区间之间的对应关系。

[0051] S304,所述控制器11合并所述第一数据和所述第二数据,并且将所述合并后的数据写入所述第三文件。

[0052] 具体的,控制器11将所述合并后的数据写入所述分配的逻辑地址区间所对应的物

理地址中。

[0053] 如果第一数据的关键字与第二数据的关键字不相同,则不需要对所述第一文件与所述第二文件进行合并。那么控制器11在机械硬盘45上创建第三文件,分别将第一文件所包含的数据和第二文件所包含的数据写入第三文件。其具体的写入第三文件的过程与将合并后的数据写入所述第三文件的过程相同,这里不再赘述。

[0054] 本发明实施例提供的文件合并方法,将合并前的文件存储在固态硬盘中,将合并后的文件存储在机械硬盘中。由于合并前的文件中的数据属于新数据,被读取的可能性较大,将其存储在固态硬盘中可以提高读数据的效率。而合并后的文件中的数据属于旧数据,被读取的可能性较小,将其存储在机械硬盘中,可以降低成本。

[0055] 在本申请所提供的几个实施例中,应该理解到,所揭露的系统,设备和方法,可以通过其它的方式实现。例如,以上所描述的装置实施例仅仅是示意性的,例如,所述模块的划分,仅仅为一种逻辑功能划分,实现时可以有另外的划分方式,例如多个模块或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口,装置或模块的间接耦合或通信连接,可以是电性,机械或其它的形式。

[0056] 所述作为分离部件说明的模块可以是或者也可以不是物理上分开的,作为模块显示的部件可以是或者也可以不是物理模块,即可以位于一个地方,或者也可以分布到多个网络模块上。可以根据实际的需要选择其中的部分或者全部模块来实现本实施例方案的目的。

[0057] 另外,在本发明各个实施例中的各功能模块可以集成在一个处理模块中,也可以是各个模块单独物理存在,也可以两个或两个以上模块集成在一个模块中。上述集成的模块既可以采用硬件的形式实现,也可以采用硬件加软件功能模块的形式实现。

[0058] 上述以软件功能模块的形式实现的集成的模块,可以存储在一个计算机可读取存储介质中。上述软件功能模块存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备等)执行本发明各个实施例所述方法的部分步骤。而前述的存储介质包括:移动硬盘、只读存储器、随机存取存储器、磁碟或者光盘等各种可以存储程序代码的介质。

[0059] 最后应说明的是:以上实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的保护范围。

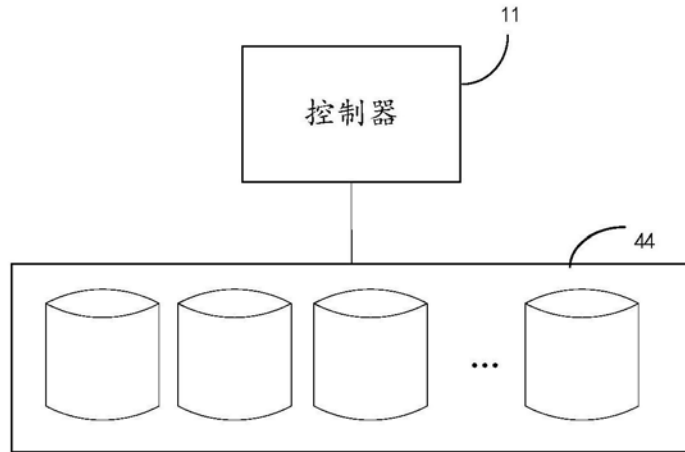


图1

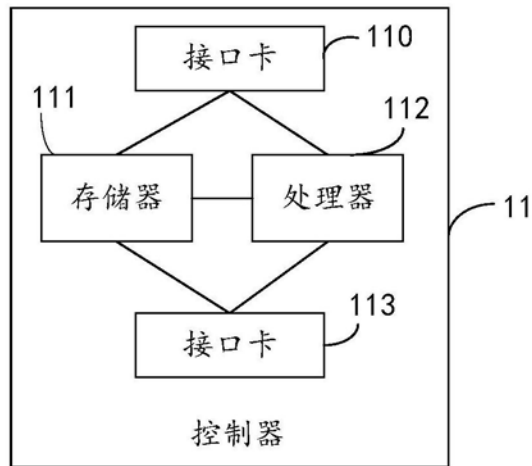


图2

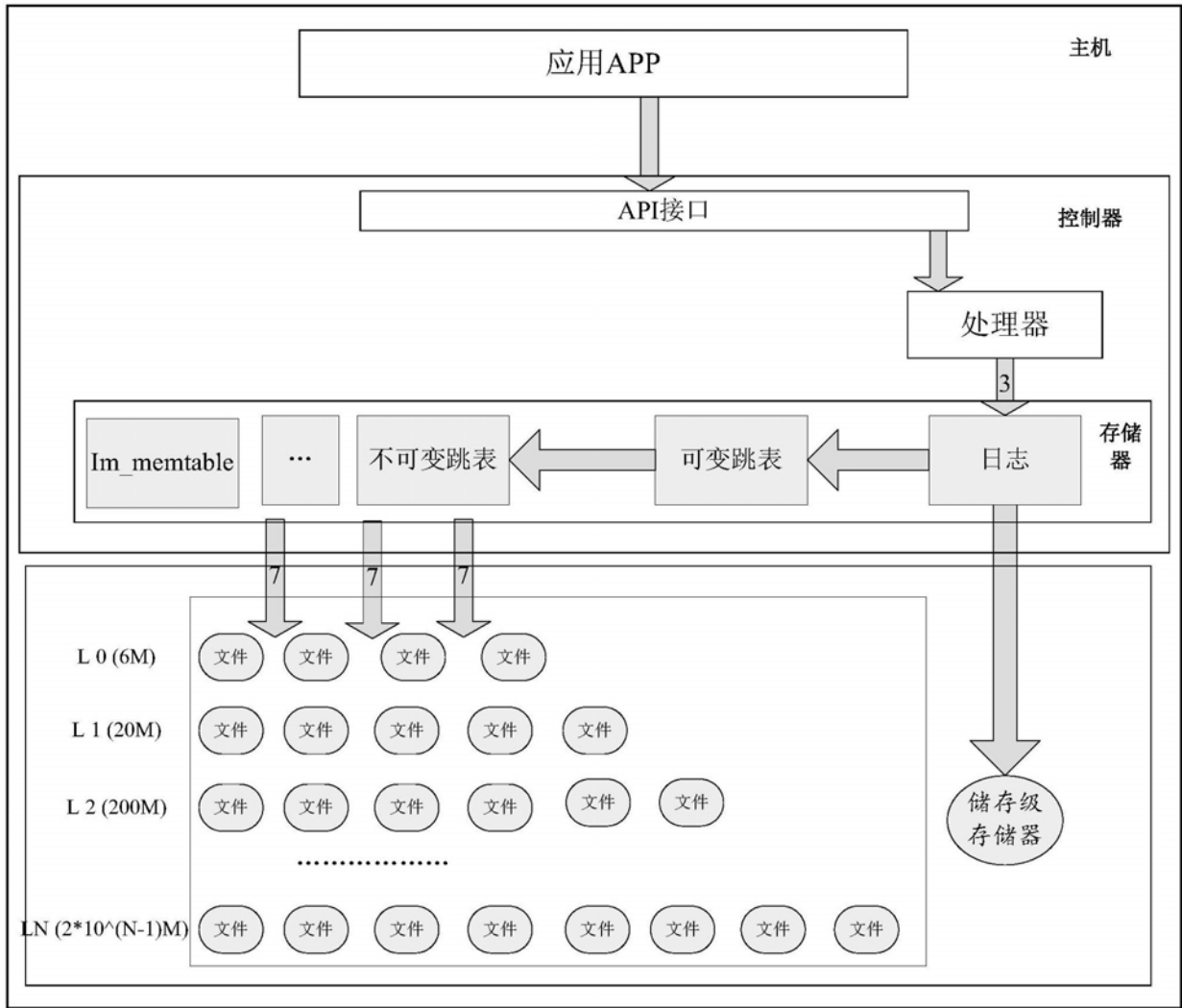


图3

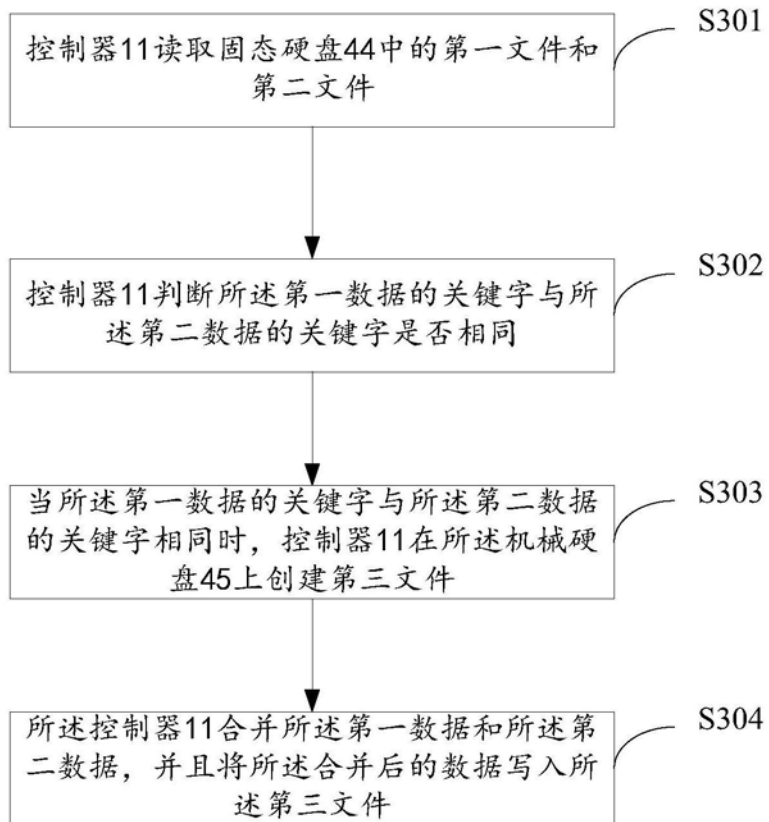


图4