(19)国家知识产权局

(12)发明专利

(10)授权公告号 CN 112486860 B
(45)授权公告日 2024.09.03

(21)申请号 201910860170.X
(22)申请日 2019.09.11
(65)同一申请的已公布的文献号
申请公布号 CN 112486860 A
(43)申请公布日 2021.03.12
(73)专利权人 伊姆西IP控股有限责任公司
地址 美国马萨诸塞州
(72)发明人 黄一帆 蔡超前
(74)专利代理机构 北京市金杜律师事务所 11256
专利代理师 王茂华 李峥宇
(51)Int.Cl.
G06F 12/1009(2016.01)

(56)对比文件
CN 101656094 A,2010.02.24
CN 108734638 A,2018.11.02

审查员 马聪聪

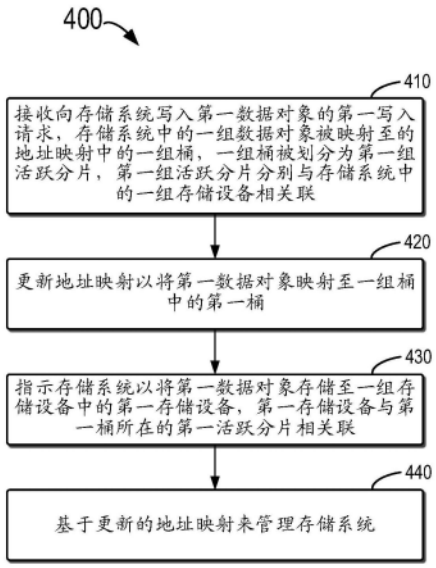
权利要求书3页 说明书12页 附图8页

(54)发明名称

管理存储系统的地址映射的方法、设备和计算机程序产品

(57)摘要

本公开涉及管理存储系统的地址映射的方法、设备和计算机程序产品。存储系统中的一组数据对象被映射至的地址映射中的一组桶,一组桶被划分为第一组活跃分片,第一组活跃分片分别与存储系统中的一组存储设备相关联。在一种方法中,接收向存储系统写入第一数据对象的第一写入请求。更新地址映射以将第一数据对象映射至一组桶中的第一桶。指示存储系统以将第一数据对象存储至一组存储设备中的第一存储设备,第一存储设备与第一桶所在的第一活跃分片相关联。基于更新的地址映射来管理存储系统。利用上述示例性实现,可以以更为高效的方式管理存储系统中的地址映射,进而提高存储系统的整体响应速度。进一步,提供了相应的设备和计算机程序产品。



CN 112486860 B

1.一种用于管理存储系统的地址映射的方法,所述存储系统中的一组数据对象被映射至的所述地址映射中的一组桶,所述一组桶被划分为第一组活跃分片,所述第一组活跃分片分别与所述存储系统中的一组存储设备相关联,所述方法包括:

接收向所述存储系统写入第一数据对象的第一写入请求;

基于所述存储系统的访问负载以及所述存储系统的负载阈值条件,调整所述第一组活跃分片中的活跃分片的分片数量,其中所述负载阈值条件包括上限阈值,并且调整所述第一组活跃分片中的活跃分片的所述分片数量包括:根据确定所述存储系统的所述访问负载高于所述上限阈值,提高所述分片;

更新所述地址映射以将所述第一数据对象映射至所述一组桶中的第一桶,产生更新的所述地址映射;

指示所述存储系统以将所述第一数据对象存储至所述一组存储设备中的第一存储设备,所述第一存储设备与所述第一桶所在的第一活跃分片相关联;以及

基于更新的所述地址映射来管理所述存储系统。

2.根据权利要求1所述的方法,其中基于更新的所述地址映射来管理所述存储系统包括:根据确定所述存储系统的状态满足预定再生条件,

将所述第一组活跃分片标识为第一组非活跃分片;以及

将所述一组桶划分至第二组活跃分片。

3.根据权利要求2所述的方法,其中所述预定再生条件包括与一个分片相关联的数据对象的阈值数量,以及所述方法进一步包括:

针对所述第一组活跃分片中的给定活跃分片,确定与所述给定活跃分片相关联的数据对象的对象数量;以及

根据确定所述对象数量高于所述阈值数量,将所述一组桶划分至第二组活跃分片。

4.根据权利要求1所述的方法,其中所述负载阈值条件包括下限阈值,以及其中调整所述第一组活跃分片中的活跃分片的所述分片数量包括:

根据确定所述存储系统的所述访问负载低于所述下限阈值,降低所述分片数量。

5.根据权利要求1所述的方法,其中基于更新的所述地址映射来管理所述存储系统包括:

确定与所述第一组活跃分片中的多个数据对象的数据分布;以及

基于所述数据分布,合并所述一组活跃分片中的前后相继的活跃分片。

6.根据权利要求2所述的方法,其中基于更新的所述地址映射来管理所述存储系统包括:

接收向所述存储系统写入第二数据对象的第二写入请求;

更新所述地址映射以将所述第二数据对象映射至所述一组桶中的第二桶;

指示所述存储系统以将所述第二数据对象存储至所述一组存储设备中的第二存储设备,所述第二存储设备与所述第二桶所在的第二活跃分片相关联。

7.根据权利要求1所述的方法,其中基于更新的所述地址映射来管理所述存储系统包括:

基于所述第一组活跃分片,确定与所述第一数据对象相关联的第一活跃分片的第一分代标识符和第一分片标识符;以及

向所述地址映射添加所述第一数据对象与所述第一活跃分片之间的第一关联关系,所述第一活跃分片利用所述第一分代标识符和所述第一分片标识符来表示。

8.根据权利要求7所述的方法,其中基于更新的所述地址映射来管理所述存储系统包括:

接收从所述存储系统读取目标数据对象的读取请求;

基于所述地址映射中包括的关联关系,确定与所述目标数据对象相关联的目标分片;以及

基于所述目标分片,从所述存储系统中读取所述目标数据对象。

9.一种用于管理存储系统的地址映射的设备,所述存储系统中的一组数据对象被映射至的所述地址映射中的一组桶,所述一组桶被划分为第一组活跃分片,所述第一组活跃分片分别与所述存储系统中的一组存储设备相关联,所述设备包括:

至少一个处理器;

易失性存储器;以及

与所述至少一个处理器耦合的存储器,所述存储器具有存储于其中的指令,所述指令在被所述至少一个处理器执行时使得所述设备执行动作,所述动作包括:

接收向所述存储系统写入第一数据对象的第一写入请求;

基于所述存储系统的访问负载以及所述存储系统的负载阈值条件,调整所述第一组活跃分片中的活跃分片的分片数量,其中所述负载阈值条件包括上限阈值,并且调整所述第一组活跃分片中的活跃分片的所述分片数量包括:根据确定所述存储系统的所述访问负载高于所述上限阈值,提高所述分片;

更新所述地址映射以将所述第一数据对象映射至所述一组桶中的第一桶,产生更新的所述地址映射;

指示所述存储系统以将所述第一数据对象存储至所述一组存储设备中的第一存储设备,所述第一存储设备与所述第一桶所在的第一活跃分片相关联;以及

基于更新的所述地址映射来管理所述存储系统。

10.根据权利要求9所述的设备,其中基于更新的所述地址映射来管理所述存储系统包括:根据确定所述存储系统的状态满足预定再生条件,

将所述第一组活跃分片标识为第一组非活跃分片;以及

将所述一组桶划分至第二组活跃分片。

11.根据权利要求10所述的设备,其中所述预定再生条件包括与一个分片相关联的数据对象的阈值数量,以及所述动作进一步包括:

针对所述第一组活跃分片中的给定活跃分片,确定与所述给定活跃分片相关联的数据对象的对象数量;以及

根据确定所述对象数量高于所述阈值数量,将所述一组桶划分至第二组活跃分片。

12.根据权利要求9所述的设备,其中所述负载阈值条件包括下限阈值,以及其中调整所述第一组活跃分片中的活跃分片的所述分片数量包括:

根据确定所述存储系统的所述访问负载低于所述下限阈值,降低所述分片数量。

13.根据权利要求9所述的设备,其中基于更新的所述地址映射来管理所述存储系统包括:

确定与所述第一组活跃分片中的多个数据对象的数据分布;以及

基于所述数据分布,合并所述一组活跃分片中的前后相继的活跃分片。

14.根据权利要求10所述的设备,其中基于更新的所述地址映射来管理所述存储系统包括:

接收向所述存储系统写入第二数据对象的第二写入请求;

更新所述地址映射以将所述第二数据对象映射至所述一组桶中的第二桶;

指示所述存储系统以将所述第二数据对象存储至所述一组存储设备中的第二存储设备,所述第二存储设备与所述第二桶所在的第二活跃分片相关联。

15.根据权利要求9所述的设备,其中基于更新的所述地址映射来管理所述存储系统包括:

基于所述第一组活跃分片,确定与所述第一数据对象相关联的第一活跃分片的第一分代标识符和第一分片标识符;以及

向所述地址映射添加所述第一数据对象与所述第一活跃分片之间的第一关联关系,所述第一活跃分片利用所述第一分代标识符和所述第一分片标识符来表示。

16.一种计算机程序产品,所述计算机程序产品被有形地存储在非瞬态计算机可读介质上并且包括机器可执行指令,所述机器可执行指令用于执行根据权利要求1-8中的任一项所述的方法。

管理存储系统的地址映射的方法、设备和计算机程序产品

技术领域

[0001] 本公开的各实现方式涉及存储系统的管理,更具体地,涉及用于管理存储系统中的地址映射的方法、设备和计算机程序产品。

背景技术

[0002] 随着数据存储技术的发展,各种数据存储设备已经能够向用户提供越来越高的数据存储能力。目前已经提出了分布式存储系统的概念,并且可以将用户数据分布在分布式存储系统中包括的各个节点上。进一步,在提高数据存储能力的同时,用户对于存储系统的响应时间也提出了越来越高的需求。目前,已经开发出了针对存储系统中存储的数据对象的地址映射建立索引以加速数据访问速度的技术方案。

[0003] 例如,可以采用分片(shading)技术来将存储系统中的存储空间划分为多个分片(shade)。该多个分片可以并行地进行诸如数据写入/读取的操作,进而提高存储系统的性能。随着存储系统中的数据量的增加,分片所涉及的数据量将越来越大并导致相关的管理工作愈加繁重。此时如何以更为有效的方式来管理地址映射并提高存储系统的性能,成为一个研究热点。

发明内容

[0004] 因而,期望能够开发并实现一种以更为有效的方式来管理存储系统的地址映射的技术方案。期望该技术方案能够与现有的存储系统相兼容,并且通过改造现有存储系统的各种配置,来以更为有效的方式管理存储系统。

[0005] 根据本公开的第一方面,提供了一种用于管理存储系统的地址映射的方法。存储系统中的一组数据对象被映射至的地址映射中的一组桶,一组桶被划分为第一组活跃分片(active shard),第一组活跃分片分别与存储系统中的一组存储设备相关联。在该方法中,接收向存储系统写入第一数据对象的第一写入请求。更新地址映射以将第一数据对象映射至一组桶中的第一桶。指示存储系统以将第一数据对象存储至一组存储设备中的第一存储设备,第一存储设备与第一桶所在的第一活跃分片相关联。基于更新的地址映射来管理存储系统。

[0006] 根据本公开的第二方面,提供了一种用于管理存储系统的地址映射的设备。存储系统中的一组数据对象被映射至的地址映射中的一组桶,一组桶被划分为第一组活跃分片,第一组活跃分片分别与存储系统中的一组存储设备相关联。该设备包括:至少一个处理器;易失性存储器;以及与至少一个处理器耦合的存储器,存储器具有存储于其中的指令,指令在被至少一个处理器执行时使得设备执行动作。该动作包括:接收向存储系统写入第一数据对象的第一写入请求;更新地址映射以将第一数据对象映射至一组桶中的第一桶;指示存储系统以将第一数据对象存储至一组存储设备中的第一存储设备,第一存储设备与第一桶所在的第一活跃分片相关联;以及基于更新的地址映射来管理存储系统。

[0007] 根据本公开的第三方面,提供了一种计算机程序产品,计算机程序产品被有形地

存储在非瞬态计算机可读介质上并且包括机器可执行指令,机器可执行指令用于执行根据本公开的第一方面的方法。

附图说明

[0008] 结合附图并参考以下详细说明,本公开各实现方式的特征、优点及其他方面将变得更加明显,在此以示例性而非限制性的方式示出了本公开的若干实现方式。在附图中:

[0009] 图1示意性示出了其中可以实现本公开的方法的存储系统的示意图;

[0010] 图2示意性示出了根据一个技术方案的存储系统的地址映射的框图;

[0011] 图3示意性示出了根据本公开的一个实现方式的用于管理存储系统的地址映射的框图;

[0012] 图4示意性示出了根据本公开的一个实现方式的用于管理存储系统的地址映射的方法的流程图;

[0013] 图5示意性示出了根据本公开的一个实现方式的调整分片数量之后的一组活跃分片的框图;

[0014] 图6示意性示出了根据本公开的一个实现方式的用于针对多个分片执行合并的框图;

[0015] 图7示意性示出了根据本公开的一个实现方式的在存储系统中查询目标数据对象的框图;以及

[0016] 图8示意性示出了根据本公开的示例性实现的用于管理存储系统的地址映射的设备的框图。

具体实施方式

[0017] 下面将参照附图更详细地描述本公开的优选实现。虽然附图中显示了本公开的优选实现,然而应该理解,可以以各种形式实现本公开而不应被这里阐述的实现所限制。相反,提供这些实现是为了使本公开更加透彻和完整,并且能够将本公开的范围完整地传达给本领域的技术人员。

[0018] 在本文中使用的术语“包括”及其变形表示开放性包括,即“包括但不限于”。除非特别申明,术语“或”表示“和/或”。术语“基于”表示“至少部分地基于”。术语“一个示例实现”和“一个实现”表示“至少一个示例实现”。术语“另一实现”表示“至少一个另外的实现”。术语“第一”、“第二”等等可以指代不同的或相同的对象。下文还可能包括其他明确的和隐含的定义。

[0019] 目前已经开发出了多种存储系统,例如,在分布式存储系统中,可以包括多个存储设备,并且可以基于地址映射来将来自用户的数据对象存储至多个存储设备中的存储设备。图1示意性示出了其中可以实现本公开的方法的存储系统的示意图100。如图1所示,存储系统110可以包括多个存储设备120、122、……、以及124。进一步,存储系统110还可以包括地址映射130,该地址映射可以将来自用户的数据对象140映射至一个或多个存储设备。

[0020] 将会理解,在此的数据对象140可以具有不同类型。例如,数据对象140可以是视频文件、音频文件、文本文件等。进一步,数据对象140还可以具有不同的大小。如果数据对象140较大(例如,高清电影),则可以将该数据对象140划分为多个较小的区块(block),并且

将多个区块分别存储至存储系统110中的多个存储设备中。此时,地址映射130可以记录数据对象140与该数据对象140的一个或多个区块之间的映射关系。换言之,地址映射130需要记录数据对象140中的每个区块被存储至存储系统110中的哪个(哪些)存储设备并且需要记录在那个(那些)存储设备的地址。

[0021] 为了便于管理,目前已经提出了分片技术,图2示意性示出了根据一个技术方案的存储系统的地址映射的框图200。如图2所示,可以基于数据对象140来获取该数据对象的哈希值。地址映射130可以包括多个分片,在此分片的数量可以依赖于存储系统110中的存储设备的数量。例如,分片的数量可以正比于存储设备的数量,并且可以为分片的数量设置最大值。

[0022] 将会理解,尽管图2中示出了一个分片对应于一个存储设备的情况,在其他示例中,分片可以对应于存储系统110中的一个或多个存储设备。在进一步的示例中,分片和存储设备之间还可以存在多对多的关系。如图2所示,针对来自存储系统110的用户的数据对象140执行哈希运算后,可以获得哈希值210。基于该哈希值210,可以确定数据对象140对应于分片220,继而可以将该数据对象140存储至与分片220相关联的存储设备122中。

[0023] 为了确保存储系统110的负载均衡,应当确保多个分片具有类似的大小,并且期望与每个分片相关联的存储设备的访问负载是相似的。然而,由于来自用户的访问存在较大波动,并且由于存储系统110中的存储设备的数量可能会出现变化,因而需要复杂的算法来确保存储系统的均衡。此时,如何以更为简单并且有效的方式来管理存储系统110的地址映射130,以便平衡用于管理各个分片的工作负载,成为一个研究热点。

[0024] 为了解决上述缺陷,本公开的实现方式提供了一种用于管理存储系统110的地址映射130的方法、设备和计算机程序产品。根据本公开的示例性实现,地址映射130可以包括一组桶(bucket),一个桶可以关联于存储系统110中的一个地址范围,例如,可以关联于一个存储设备中的一个地址范围。桶的数量可以依赖于存储系统110中的存储空间的大小,并且可以为桶的数量设置最大值。例如,可以将整个存储系统110中的存储空间划分至2^m个桶(例如,当m=7时,则存在128个桶)。

[0025] 可以为一组桶设置标识符(例如,0至127),可以基于数据对象140的哈希值来将该数据对象140映射至相应的桶。例如,某个数据对象的哈希值为11,则可以将该数据对象映射至第11个桶。多个数据对象可以被映射至相同或者不同的桶。随着存储系统110的运行,各个桶中包括的数据对象的数量将会增加。可以将一组桶划分为一组分片,例如可以将2^m个桶划分为n个分片。假设m=7并且n=3,则可以将128个桶划分为3个分片。

[0026] 为方便描述起见,在下文中将仅以数据对象140仅包括一个区块的情况进行描述。当数据对象140包括较多数据时,可以将数据对象140划分至多个区块。此时,针对数据对象140中的每个区块的处理是相似的。图3示意性示出了根据本公开的一个实现方式的用于管理存储系统110的地址映射130的框图300。如图3所示,可以以环形方式来管理多个桶。当以较为平均的方式进行划分时,可以将128个桶划分为3个分片。分片310例如可以包括第0至43个桶,分片312可以包括第44至第86个桶,并且分片314可以包括第87至第127个桶。将会理解,在此划分桶的方式仅仅是示意性的,还可以按照其他方式进行划分。

[0027] 根据本公开的示例性实现方式,提出了活跃分片的概念。在此的活跃分片是指地址映射130中的可以被改变的分片,换言之,可以通过将新的数据对象映射至活跃分片中的

桶,来更新地址映射130。如图3所示,在接收到向存储系统110写入新的数据对象的写入请求后,可以基于数据对象的哈希值来将该数据对象映射至活跃的分片310中的桶320。相对于活跃分片而言,地址映射130还可以包括非活跃(inactive)分片。在此非活跃分片是指地址映射130中的不能被改变的分片。换言之,不能向非活跃分片中的桶映射新的数据对象,而是仅能针对非活跃分片执行查询操作。

[0028] 此时,可以指示存储系统110以将数据对象140存储至一组存储设备中的第一存储设备,第一存储设备与第一桶所在的第一活跃分片相关联。继而,可以基于更新的地址映射来管理存储系统110。在下文中,将参见图4描述有关管理存储系统110的地址映射130的更多细节。

[0029] 图4示意性示出了根据本公开的一个实现方式的用于管理存储系统110的地址映射130的方法400的流程图。将会理解,在此的存储系统110中的一组数据对象被映射至的地址映射中的一组桶,一组桶被划分为第一组活跃分片,第一组活跃分片分别与存储系统中的一组存储设备相关联。如图4中的框410所示,可以接收向存储系统110写入第一数据对象的第一写入请求。在此的第一数据对象即为将要向存储系统中写入的新的数据对象。将会理解,下文中仅以示例方式来描述针对一个写入请求的操作。根据本公开的示例性实现方式,存储系统110可以接收到一个或多个写入请求,同时还可以接收到一个或多个读取请求。此时,分别基于地址映射中的不同分片来服务于这些请求:可以利用活跃分片来服务于写入请求,可以利用活跃分片和非活跃分片两者来服务于读取请求。

[0030] 在框420处,可以更新地址映射130以将第一数据对象映射至一组桶中的第一桶。在此,可以针对第一数据对象执行哈希运算,以便确定该第一数据对象的哈希值。例如,可以采用预定的哈希函数,并基于第一数据对象的标识符来获取哈希值210。又例如,还可以基于第一数据对象的标识符、所有者的用户标识符、数据对象中的数据、以及时间戳等来生成相应的哈希值。

[0031] 在框430处,可以指示存储系统110以将第一数据对象存储至一组存储设备中的第一存储设备。在此,第一存储设备与第一桶所在的第一活跃分片相关联。换言之,可以将与第一数据对象相关联的信息写入至活跃分片中的桶。例如,如果第一数据对象的哈希值为11,由于0＜11＜43,则可以将第一数据对象存储至与分片310相关联的存储设备。又例如,例如,如果第一数据对象的哈希值为57,由于44＜57＜86,则可以将第一数据对象存储至与分片312相关联的存储设备。例如,如果第一数据对象的哈希值为93,由于87＜93＜127,则可以将第一数据对象存储至与分片314相关联的存储设备。此时,图3中的分片310、312和314均为活跃分片。

[0032] 在框440处,可以基于更新的地址映射130来管理存储系统110。将会理解,更新的地址映射130中已经包括了刚刚被存储至存储系统110中的第一数据对象的信息。基于更新的地址映射,可以确定从存储系统110中的哪个地址读取刚刚被存储的数据对象。在存储系统110的操作期间,如果接收到向存储系统110写入其他数据对象的写入请求,还可以进一步更新当前的地址映射130,以便向该地址映射130中添加将被写入的数据对象的信息。

[0033] 将会理解,在存储系统110的初始运行阶段,地址映射130可以仅包括一组活跃分片,随着存储系统110的运行,每个活跃分片中的桶中将涉及越来越多的数据对象。由于管理分片所涉及的资源将会增加,此时可以通过再生(regeneration)操作来生成其他分片,

以便管理向存储系统110写入新的数据对象的写入操作。根据本公开的示例性实现方式,还可以将当前的第一组活跃分片转换为非活跃分片。在此的非活跃分片可以用于管理从存储系统110中读取数据对象的读取操作。利用本公开的示例性实现方式,可以将用于管理不同分片的操作分配至不同的设备(例如,可以由不同存储设备中的处理器来执行处理)。因而,可以以更为并行并且有效的方式来管理地址映射130。

[0034] 根据本公开的示例性实现方式,如果确定存储系统110的状态满足预定再生条件,则可以将第一组活跃分片标识为第一组非活跃分片。例如,可以利用第一分代标识符,来表示多个活跃分片所属于的分代。假设在存储系统110的初始启动期间设置了一组活跃分片,则可以向该组活跃分片赋予相应的分代标识符(例如,以“Generation01”表示)。

[0035] 进一步,可以为每个分片赋予分片标识符,例如,可以利用“Shard01”表示第一个分片,利用“Shard02”表示第二个分片,等等。此时,对于每个分片而言,可以以二元组(分代标识符,分片标识符)来唯一地表示每个分片。例如,(Generation01,Shard01)可以表示第一分代中的第一个分片。将会理解,尽管上文中以分代标识符为示例描述了如何区分各个分代,根据本公开的示例性实现方式,还可以基于其他方式来进行区分。例如,可以基于数据对象的时间戳来实现。

[0036] 随着存储系统110的运行,当满足预定再生条件时,可以将初始的一组活跃分片标记为非活跃状态,并进一步生成一组活跃分片。例如,可以为新生成的一组活跃分片赋予分代标识符“Generation02”。此时,新生成的一组活跃分片中的各个分片可以被分别表示为(Generation02,Shard01)、(Generation02,Shard02)、(Generation02,Shard03)。可以为当前处于活跃状态的一组分片设置标识符,备选地和/或附加地,还可以定义具有最大分代标识符的一组分片处于活跃状态,而具有较小分代标识符的其他一组或者多组分片处于非活跃状态。

[0037] 在存储系统110的运行过程中,当存储系统110接收到写入请求时,可以将待写入数据对象映射至当前的一组活跃分片中的桶,并且向存储系统110中的与该桶相关联的存储设备写入待写入的数据对象。换言之,此时可以更新地址映射130中的一组活跃分片,以便服务于写入请求。当存储系统110接收到读取请求时,可以查询地址映射130中的全部分片(包括活跃分片和非活跃分片),以便找到与待读取数据对象相关联的分片,进而从与找到的分片相对应的存储设备中读取数据。

[0038] 将会理解,随着存储系统的运行,越来越多的数据对象将被写入至存储系统的一组存储设备中。这使得每个活跃分片将涉及越来越多的数据对象。随着与一个分片相关联的数据对象的数量不断增加,管理该分片涉及的资源的数量也将增大,并且管理的复杂度也将会提高。根据本公开的示例性实现方式,预定再生条件可以包括与一个分片相关联的数据对象的数量超过的阈值数量。

[0039] 具体地,在存储系统110的运行期间,可以统计每个活跃分片中所涉及的数据对象的数量。例如,针对第一组活跃分片中的给定活跃分片,可以在预定时间间隔确定与给定活跃分片相关联的数据对象的对象数量。备选地和/或附加地,还可以在每次将要更新活跃分片时进行统计、或者还可以针对每个活跃分片设置专用计数器来存储该活跃分片所涉及的数据对象的数量。可以根据确定对象数量与阈值数量之间的关系来确定是否执行再生。如果确定的数量高于阈值数量,可以执行再生操作,此时可以将当前的一组活跃分片标识为

非活跃分片,并且将一组桶划分至新的第二组活跃分片。

[0040] 继续图3的示例,在三个分片310、312和314中,如果分片310的所涉及的数据对象的数量高于预定的阈值数量,则可以执行再生操作。假设三个分片310、312和314的标识符分别为(Generation01,Shard01)、(Generation01,Shard01)、(Generation01,Shard03),则可以生成新的一组活跃分片(Generation02,Shard01)、(Generation02,Shard01)、(Generation02,Shard03)。此时,可以分配不同的处理器来用于管理不同组的分片。例如,可以分配存储设备120中的处理器来管理分代标识符为"Generation01"的分片,可以分配存储设备122中的处理器来管理分代标识符为"Generation02"的分片。根据本公开的示例性实现方式,还可以分配不同的处理器来管理具有相同分代标识符的不同分片。利用本公开的示例性实现方式,可以更为充分地利用存储系统110中的处理资源,以便以更为并行的方式来管理存储系统110的地址映射130。

[0041] 将会理解,上文中仅仅示意性示出了以分片所涉及的数据对象的数量来作为再生条件的示例。根据本公开的示例性实现方式,再生条件还可以涉及更多的示例,只要再生条件可以指示用于管理该分片所需的处理资源相关联的指标。例如,再生条件可以包括分片所涉及数据对象的数据量,分片本身所占用的数据量,等等。

[0042] 将会理解,随着存储系统110的运行,存储系统110所接收到的访问请求(包括读取/写入请求)的频率也可以不断变化。例如,在一段时间内存储系统110可能会频繁地接收到写入请求,此时大量的数据对象将被映射至活跃分片,这将导致管理该活跃分片的工作负载急剧上升。又例如,在一段时间内存储系统110可能会仅接收到较少的写入请求甚至没有接收到任何写入请求,此时管理某个活跃分片的工作负载将会保持在较低水平。

[0043] 鉴于上述情况,根据本公开的示例性实现方式,可以为活跃分片设置阈值负载。具体地,该负载阈值可以指示活跃分片在预定时间段内可以处理的请求的推荐数量。该负载阈值可以包括上限阈值和下限阈值,以便指示推荐数量的范围。例如,负载阈值可以被定义为在活动分片每秒可以处理的请求的数量的范围为(low,high)。此时,可以首先统计存储系统110的访问负载,并基于访问负载与上限阈值和下限阈值之间的关系,确定如何调整活动分片的数量。

[0044] 根据本公开的示例性实现方式,可以统计存储系统110在最近一段时间内所接收到的写入请求的数量,并且基于该数量来确定存储系统110的访问负载。如果访问负载高于上限阈值high,则表示当前一组活跃分片正在面临较高的工作负载。因而,应当提高当前活跃分片的数量。例如,可以基于预定步长来递增当前活跃分片的数量。根据本公开的示例性实现方式,如果存储系统110的访问负载低于下限阈值low,则表示当前一组活跃分片正在面临较低的工作负载。因而,应当降低当前活跃分片的数量。例如,可以基于预定步长来递减当前活跃分片的数量。在下文中,将参见图5描述调整当前活跃分片的数量的更多细节。

[0045] 图5示意性示出了根据本公开的一个实现方式的调整分片数量之后的一组活跃分片的框图500。继续上文图3的示例,如果确定图3所示的3个活跃分片的相关的工作负载较高,则可以调整活跃分片的数量。例如,可以将活跃分片的数量调整至5个。如图5所示,一组桶可以被划分为5个活跃分片:分片510可以包括第0至26个桶,分片512可以包括第27至52个桶,分片514可以包括第53至77个桶,分片516可以包括第78至103个桶,分片518可以包括第104至127个桶。将会理解,尽管图5仅示意性示出了将当前活跃分片的数量调整至5的示

例,根据本公开的示例性实现方式,还可以基于存储系统110的访问负载和上限阈值/下限阈值之间的关系,确定调整后的目标数量。

[0046] 根据本公开的示例性实现方式,假设当前存在N个活跃分片,一个活跃分片可以处理的访问请求的数量的范围为(low,high),并且第i个分片需要处理的访问请求的数量为s_i。当确定访问负载较高并且需要提高活跃分片的数量时,可以基于如下公式1来确定目标数量Num_{high}:

[0047] $$Num_{high} = \left\lceil \frac{\sum_{i=0}^{N-1} s_i}{high} \times N \right\rceil \qquad \text{公式 1}$$

[0048] 在公式1中,Num_{high}表示调整后的活动分片的数量,N表示调整前的活动分片的数量,high表示一个活跃分片可以处理的访问请求的上限,s_i表示第i个分片需要处理的访问请求的数量。

[0049] 类似地,当确定访问负载较低并且需要降低活跃分片的数量时,可以基于如下公式2来确定目标数量Num_{low}:

[0050] $$Num_{low} = \left\lceil \frac{\sum_{i=0}^{N-1} s_i}{low} \times N \right\rceil \qquad \text{公式 2}$$

[0051] 在公式2中,Num_{low}表示调整后的活动分片的数量,N表示调整前的活动分片的数量,low表示一个活跃分片可以处理的访问请求的下限,s_i表示第i个分片需要处理的访问请求的数量。

[0052] 随着存储系统110的运行,活跃分片的数量可以被调整至不同的数值,这可能会导致存在较小的分片(称为“碎片分片”)。因而,可以将这些碎片分片进行合并,以形成较大的分片以便于在存储系统110中管理各个分片。

[0053] 根据本公开的示例性实现方式,确定与第一组活跃分片中的相应活跃分片相关联的多个数据对象的相应数据分布。进一步,可以基于相应数据分布,合并一组活跃分片中的前后相继的活跃分片。在下文中,将参见下文表1所示的伪代码来描述有关合并碎片的更多细节。

[0054]

表 1 合并碎片的示例操作

```
def merge():
sort_by_key(sealed shards);
iterate through the sorted list:
    current_key_range ← key_range(current_shard);
    subSum ← Sum(current_shard) + subSum;
  subRange = combine_range(current_key_range);
    if subSum > low:
        archived_shard ← archive(subRange, current_shards);
      subSum ← 0;
      subRange ← [];
  else if reach the end of list:
        archived_shard ← archive(subRange, current_shards);
mark merge complete and regenerate
```

[0055] 按照如上文表1所示,可以将同一个分代中的碎片分片按照环上的分布进行排序,

然后将相邻的小分片合并成接近目标大小的较大分片。在下文中,将参见图6描述有关合并的更多细节。图6示意性示出了根据本公开的一个实现方式的用于针对多个分片执行合并的框图600。如图6所示,假设存在多个分片610、620等,则可以基于各个分片在环上的分布,来将前后相邻的分片合并至较大的分片。例如,可以将分片610和分配620合并至较大的分片。

[0056] 根据本公开的示例性实现方式,可以不断地向存储系统110中写入数据对象。可以接收向存储系统110写入第二数据对象的第二写入请求;可以更新地址映射以将第二数据对象映射至一组桶中的第二桶;并且可以指示存储系统以将第二数据对象存储至一组存储设备中的第二存储设备,第二存储设备与第二桶所在的第二活跃分片相关联。在此,可以按照与上文描述的方法400类似的方式来处理第一写入请求。

[0057] 根据本公开的示例性实现方式,对于存储系统110中的每个被存储的数据对象,可以在地址映射130中存储数据对象的分代标识符。例如,对于特定数据对象A而言,在向存储系统110中的一个存储设备写入该数据对象A时,假设利用第一分代的活跃分片来管理该数据对象A,则该数据对象A的分代标识符可以是“Generation01”。以此方式,可以便于在接收到针对数据对象A的读取请求时,迅速找到该数据对象A的地址映射所在的分代,进而找到相应的分片信息。

[0058] 将会理解,地址映射130中可以包括仅一组活跃分片,并且可以包括一组或者多组非活跃分片。此时活跃分片用于服务于写入请求,而活跃分片和非活跃分片两者用于服务于读取请求。此时,管理多个分片的任务可以被分配至不同的处理器。利用本公开的示例性实现方式,可以尽可能地利用存储系统中的多个处理器来并行地服务于多个写入/读取请求。在下文中,将参见图7描述如何从存储系统110中读取数据。

[0059] 图7示意性示出了根据本公开的一个实现方式的在存储系统110中读取目标数据对象的框图700。根据本公开的示例性实现方式,可以为每个分代的分片设置相应的标识符。例如,可以采用分代标识符732和分片标识符734来唯一地标识每个分片730。此时,地址映射130可以包括第一分代的分片710(处于非激活状态)以及第二分代的分片720(处于激活状态)。将会理解,尽管图7仅示意性示出了包括一组非活跃分片和一组活跃分片的情况,根据本公开的示例性实现方式,还可以包括多组非活跃分片和一组活跃分片。

[0060] 根据本公开的示例性实现方式,可以接收从存储系统110读取目标数据对象的读取请求。基于地址映射130中包括的关联关系,确定与目标数据对象相关联的目标分片。基于目标分片,从存储系统110中读取目标数据对象。

[0061] 具体地,在接收到针对存储系统110的读取请求之后,可以基于读取请求中指定的将要被读取的数据对象的标识符来确定该数据对象的哈希值。继而,还可以首先确定该数据对象所属于的分代,假设哈希值为11的数据对象属于分代“Generation01”,则可以在分代标识符为“Generation01”的一组分片中查找与该数据对象相对应的分片。进而,可以基于找到的分片来确定关于数据对象的索引信息,进而从存储设备110中检索到数据对象。又例如,假设哈希值为11的数据对象属于分代“Generation02”,则可以在分代标识符为“Generation02”的一组分片中查找与该数据对象相对应的分片。

[0062] 在上文中已经参见图2至图7详细描述了根据本公开的方法400的示例,在下文中将描述相应的装置的实现。根据本公开的示例性实现,提供了一种用于管理存储系统的地

址映射的装置。存储系统中的一组数据对象被映射至的地址映射中的一组桶,一组桶被划分为第一组活跃分片,第一组活跃分片分别与存储系统中的一组存储设备相关联。该装置包括:接收模块,配置用于接收向存储系统写入第一数据对象的第一写入请求;更新模块,配置用于更新地址映射以将第一数据对象映射至一组桶中的第一桶;指示模块,配置用于指示存储系统以将第一数据对象存储至一组存储设备中的第一存储设备,第一存储设备与第一桶所在的第一活跃分片相关联;以及管理模块,配置用于模块,配置用于基于更新的地址映射来管理存储系统。

[0063] 根据本公开的示例性实现方式,管理模块包括:再生模块,配置用于根据确定存储系统的状态满足预定再生条件,将第一组活跃分片标识为第一组非活跃分片;以及划分模块,配置用于将一组桶划分至第二组活跃分片。

[0064] 根据本公开的示例性实现方式,再生条件包括与一个分片相关联的数据对象的阈值数量,以及划分模块进一步包括:确定模块,配置用于针对第一组活跃分片中的给定活跃分片,确定与给定活跃分片相关联的数据对象的对象数量;以及调整模块,配置用于根据确定对象数量高于阈值数量,将一组桶划分至第二组活跃分片。

[0065] 根据本公开的示例性实现方式,该装置进一步包括:负载模块,配置用于基于存储系统的访问负载以及存储系统的负载阈值条件,调整第一组活跃分片中的活跃分片的分片数量。

[0066] 根据本公开的示例性实现方式,负载阈值条件包括下限阈值,以及调整模块包括:降低模块,配置用于根据确定存储系统的访问负载低于下限阈值,降低分片数量。

[0067] 根据本公开的示例性实现方式,负载阈值条件包括上限阈值,以及调整模块包括:提高模块,配置用于根据确定存储系统的访问负载高于上限阈值,提高分片。

[0068] 根据本公开的示例性实现方式,管理模块包括:分布确定模块,配置用于确定与第一组活跃分片中的多个数据对象的数据分布;以及合并模块,配置用于基于数据分布,合并一组活跃分片中的前后相继的活跃分片。

[0069] 根据本公开的示例性实现方式,接收模块进一步配置用于接收向存储系统写入第二数据对象的第二写入请求;更新模块进一步配置用于更新地址映射以将第二数据对象映射至一组桶中的第二桶;指示模块进一步配置用于指示存储系统以将第二数据对象存储至一组存储设备中的第二存储设备,第二存储设备与第二桶所在的第二活跃分片相关联。

[0070] 根据本公开的示例性实现方式,管理模块包括:标识符确定模块,配置用于基于第一组活跃分片,确定与第一数据对象相关联的第一活跃分片的第一分代标识符和第一分片标识符;以及添加模块,配置用于向地址映射添加第一数据对象与第一活跃分片之间的第一关联关系,第一活跃分片利用第一分代标识符和第一分片标识符来表示。

[0071] 根据本公开的示例性实现方式,接收模块进一步配置用于接收从存储系统读取目标数据对象的读取请求;分片确定模块,配置用于基于地址映射中包括的关联关系,确定与目标数据对象相关联的目标分片;以及读取模块,配置用于基于目标分片,从存储系统中读取目标数据对象。

[0072] 图8示意性示出了根据本公开的示例性实现的用于管理存储系统的地址映射的设备800的框图。如图所示,设备800包括中央处理单元(CPU)801,其可以根据存储在只读存储器(ROM)802中的计算机程序指令或者从存储单元808加载到随机访问存储器(RAM)803中的

计算机程序指令,来执行各种适当的动作和处理。在RAM 803中,还可存储设备800操作所需的各种程序和数据。CPU 801、ROM802以及RAM 803通过总线804彼此相连。输入/输出(I/O)接口805也连接至总线804。

[0073] 设备800中的多个部件连接至I/O接口805,包括:输入单元806,例如键盘、鼠标等;输出单元807,例如各种类型的显示器、扬声器等;存储单元808,例如磁盘、光盘等;以及通信单元809,例如网卡、调制解调器、无线通信收发机等。通信单元809允许设备800通过诸如因特网的计算机网络和/或各种电信网络与其他设备交换信息/数据。

[0074] 上文所描述的各个过程和处理,例如方法400,可由处理单元801执行。例如,在一些实现中,方法400可被实现为计算机软件程序,其被有形地包含于机器可读介质,例如存储单元808。在一些实现中,计算机程序的部分或者全部可以经由ROM 802和/或通信单元809而被载入和/或安装到设备800上。当计算机程序被加载到RAM 803并由CPU 801执行时,可以执行上文描述的方法400的一个或多个步骤。备选地,在其他实现中,CPU 801也可以以其他任何适当的方式被配置以实现上述过程/方法。

[0075] 根据本公开的示例性实现,提供了一种用于管理存储系统的地址映射的设备,存储系统中的一组数据对象被映射至的地址映射中的一组桶,一组桶被划分为第一组活跃分片,第一组活跃分片分别与存储系统中的一组存储设备相关联。该设备包括:至少一个处理器;易失性存储器;以及与至少一个处理器耦合的存储器,存储器具有存储于其中的指令,指令在被至少一个处理器执行时使得设备执行动作。该动作包括:接收向存储系统写入第一数据对象的第一写入请求;更新地址映射以将第一数据对象映射至一组桶中的第一桶;指示存储系统以将第一数据对象存储至一组存储设备中的第一存储设备,第一存储设备与第一桶所在的第一活跃分片相关联;以及基于更新的地址映射来管理存储系统。

[0076] 根据本公开的示例性实现方式,基于更新的地址映射来管理存储系统包括:根据确定存储系统的状态满足预定再生条件,将第一组活跃分片标识为第一组非活跃分片;以及将一组桶划分至第二组活跃分片。

[0077] 根据本公开的示例性实现方式,预定再生条件包括与一个分片相关联的数据对象的阈值数量,以及该设备进一步包括:针对第一组活跃分片中的给定活跃分片,确定与给定活跃分片相关联的数据对象的对象数量;以及根据确定对象数量高于阈值数量,将一组桶划分至第二组活跃分片。

[0078] 根据本公开的示例性实现方式,该设备进一步包括:基于存储系统的访问负载以及存储系统的负载阈值条件,调整第一组活跃分片中的活跃分片的分片数量。

[0079] 根据本公开的示例性实现方式,负载阈值条件包括下限阈值,以及其中调整第一组活跃分片中的活跃分片的分片数量包括:根据确定存储系统的访问负载低于下限阈值,降低分片数量。

[0080] 根据本公开的示例性实现方式,负载阈值条件包括上限阈值,以及其中调整第一组活跃分片中的活跃分片的分片数量包括:根据确定存储系统的访问负载高于上限阈值,提高分片。

[0081] 根据本公开的示例性实现方式,基于更新的地址映射来管理存储系统包括:确定与第一组活跃分片中的多个数据对象的数据分布;以及基于数据分布,合并一组活跃分片中的前后相继的活跃分片。

[0082] 根据本公开的示例性实现方式,基于更新的地址映射来管理存储系统包括:接收向存储系统写入第二数据对象的第二写入请求;更新地址映射以将第二数据对象映射至一组桶中的第二桶;指示存储系统以将第二数据对象存储至一组存储设备中的第二存储设备,第二存储设备与第二桶所在的第二活跃分片相关联。

[0083] 根据本公开的示例性实现方式,基于更新的地址映射来管理存储系统包括:基于第一组活跃分片,确定与第一数据对象相关联的第一活跃分片的第一分代标识符和第一分片标识符;以及向地址映射添加第一数据对象与第一活跃分片之间的第一关联关系,第一活跃分片利用第一分代标识符和第一分片标识符来表示。

[0084] 根据本公开的示例性实现方式,基于更新的地址映射来管理存储系统包括:接收从存储系统读取目标数据对象的读取请求;基于地址映射中包括的关联关系,确定与目标数据对象相关联的目标分片;以及基于目标分片,从存储系统中读取目标数据对象。

[0085] 根据本公开的示例性实现,提供了一种计算机程序产品,计算机程序产品被有形地存储在非瞬态计算机可读介质上并且包括机器可执行指令,机器可执行指令用于执行根据本公开的方法。

[0086] 根据本公开的示例性实现,提供了一种计算机可读介质。计算机可读介质上存储有机器可执行指令,当机器可执行指令在被至少一个处理器执行时,使得至少一个处理器实现根据本公开方法。

[0087] 本公开可以是方法、设备、系统和/或计算机程序产品。计算机程序产品可以包括计算机可读存储介质,其上载有用于执行本公开的各个方面的计算机可读程序指令。

[0088] 计算机可读存储介质可以是可以保持和存储由指令执行设备使用的指令的有形设备。计算机可读存储介质例如可以是——但不限于——电存储设备、磁存储设备、光存储设备、电磁存储设备、半导体存储设备或者上述的任意合适的组合。计算机可读存储介质的更具体的例子(非穷举的列表)包括:便携式计算机盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、静态随机存取存储器(SRAM)、便携式压缩盘只读存储器(CD-ROM)、数字多功能盘(DVD)、记忆棒、软盘、机械编码设备、例如其上存储有指令的打孔卡或凹槽内凸起结构、以及上述的任意合适的组合。这里所使用的计算机可读存储介质不被解释为瞬时信号本身,诸如无线电波或者其他自由传播的电磁波、通过波导或其他传输媒介传播的电磁波(例如,通过光纤电缆的光脉冲)、或者通过电线传输的电信号。

[0089] 这里所描述的计算机可读程序指令可以从计算机可读存储介质下载到各个计算/处理设备,或者通过网络、例如因特网、局域网、广域网和/或无线网下载到外部计算机或外部存储设备。网络可以包括铜传输电缆、光纤传输、无线传输、路由器、防火墙、交换机、网关计算机和/或边缘服务器。每个计算/处理设备中的网络适配卡或者网络接口从网络接收计算机可读程序指令,并转发该计算机可读程序指令,以供存储在各个计算/处理设备中的计算机可读存储介质中。

[0090] 用于执行本公开操作的计算机程序指令可以是汇编指令、指令集架构(ISA)指令、机器指令、机器相关指令、微代码、固件指令、状态设置数据、或者以一种或多种编程语言的任意组合编写的源代码或目标代码,编程语言包括面向对象的编程语言—诸如Smalltalk、C++等,以及常规的过程式编程语言—诸如“C”语言或类似的编程语言。计算机可读程序指

令可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中,远程计算机可以通过任意种类的网络—包括局域网(LAN)或广域网(WAN)—连接到用户计算机,或者,可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。在一些实现中,通过利用计算机可读程序指令的状态信息来个性化定制电子电路,例如可编程逻辑电路、现场可编程门阵列(FPGA)或可编程逻辑阵列(PLA),该电子电路可以执行计算机可读程序指令,从而实现本公开的各个方面。

[0091] 这里参照根据本公开实现的方法、装置(系统)和计算机程序产品的流程图和/或框图描述了本公开的各个方面。应当理解,流程图和/或框图的每个方框以及流程图和/或框图中各方框的组合,都可以由计算机可读程序指令实现。

[0092] 这些计算机可读程序指令可以提供给通用计算机、专用计算机或其他可编程数据处理装置的处理单元,从而生产出一种机器,使得这些指令在通过计算机或其他可编程数据处理装置的处理单元执行时,产生了实现流程图和/或框图中的一个或多个方框中规定的功能/动作的装置。也可以把这些计算机可读程序指令存储在计算机可读存储介质中,这些指令使得计算机、可编程数据处理装置和/或其他设备以特定方式工作,从而,存储有指令的计算机可读介质则包括一个制造品,其包括实现流程图和/或框图中的一个或多个方框中规定的功能/动作的各个方面的指令。

[0093] 也可以把计算机可读程序指令加载到计算机、其他可编程数据处理装置、或其他设备上,使得在计算机、其他可编程数据处理装置或其他设备上执行一系列操作步骤,以产生计算机实现的过程,从而使得在计算机、其他可编程数据处理装置、或其他设备上执行的指令实现流程图和/或框图中的一个或多个方框中规定的功能/动作。

[0094] 附图中的流程图和框图显示了根据本公开的多个实现的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段或指令的一部分,模块、程序段或指令的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个连续的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意的是,框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合,可以用执行规定的功能或动作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0095] 以上已经描述了本公开的各实现,上述说明是示例性的,并非穷尽性的,并且也不限于所公开的各实现。在不偏离所说明的各实现的范围和精神的情况下,对于本技术领域的普通技术人员来说许多修改和变更都是显而易见的。本文中所用术语的选择,旨在最好地解释各实现的原理、实际应用或对市场中的技术的改进,或者使本技术领域的其他普通技术人员能理解本文公开的各实现。

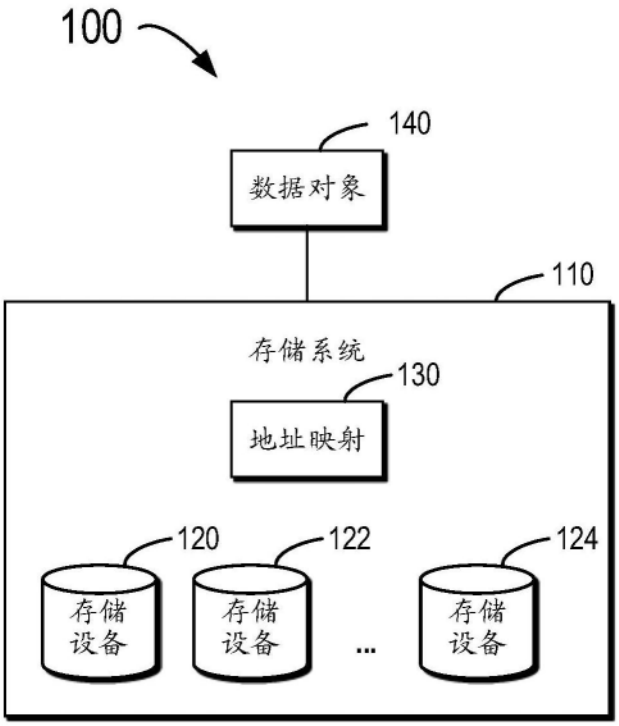


图1

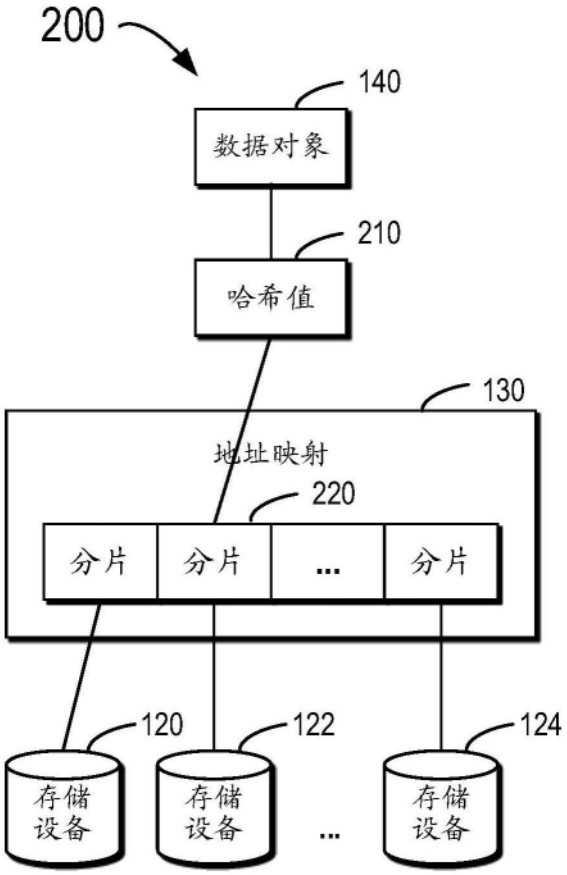

200
140
数据对象
210
哈希值
130
地址映射
220
分片
分片
...
分片
120
122
124
存储设备
存储设备
...
存储设备

图2

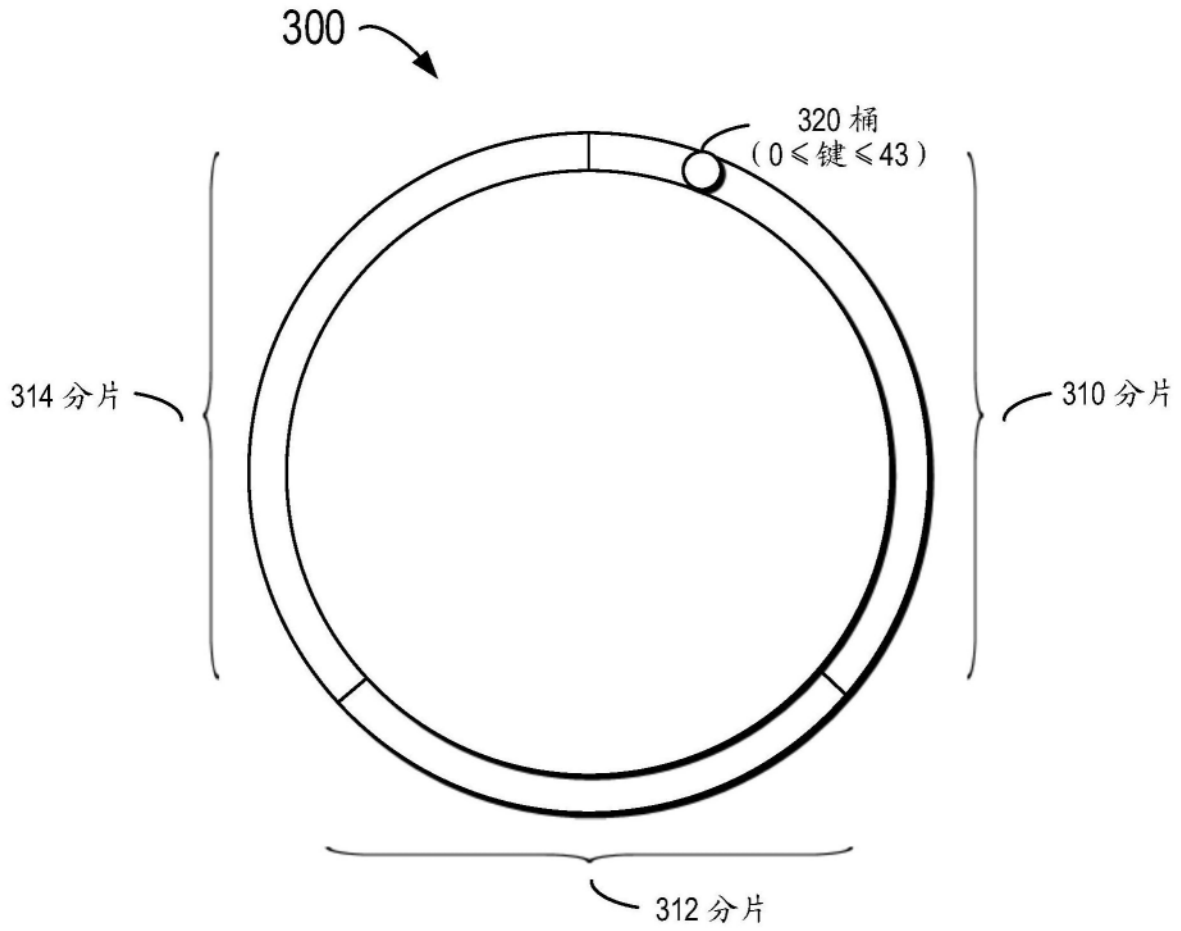


图3

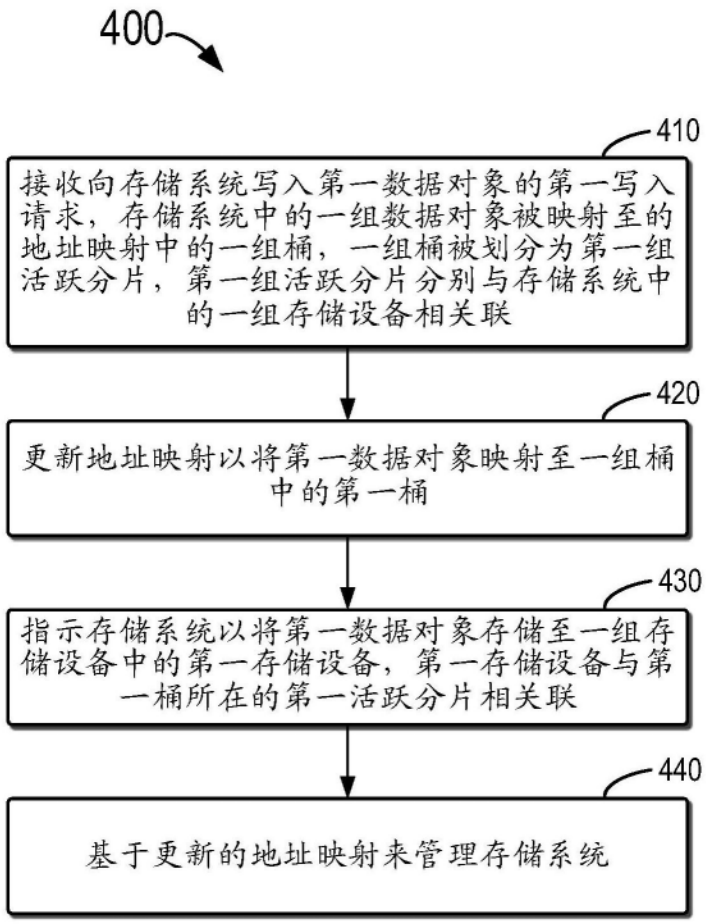

400
410
接收向存储系统写入第一数据对象的第一写入请求，存储系统中的一组数据对象被映射至的地址映射中的一组桶，一组桶被划分为第一组活跃分片，第一组活跃分片分别与存储系统中的一组存储设备相关联
420
更新地址映射以将第一数据对象映射至一组桶中的第一桶
430
指示存储系统以将第一数据对象存储至一组存储设备中的第一存储设备，第一存储设备与第一桶所在的第一活跃分片相关联
440
基于更新的地址映射来管理存储系统

图4

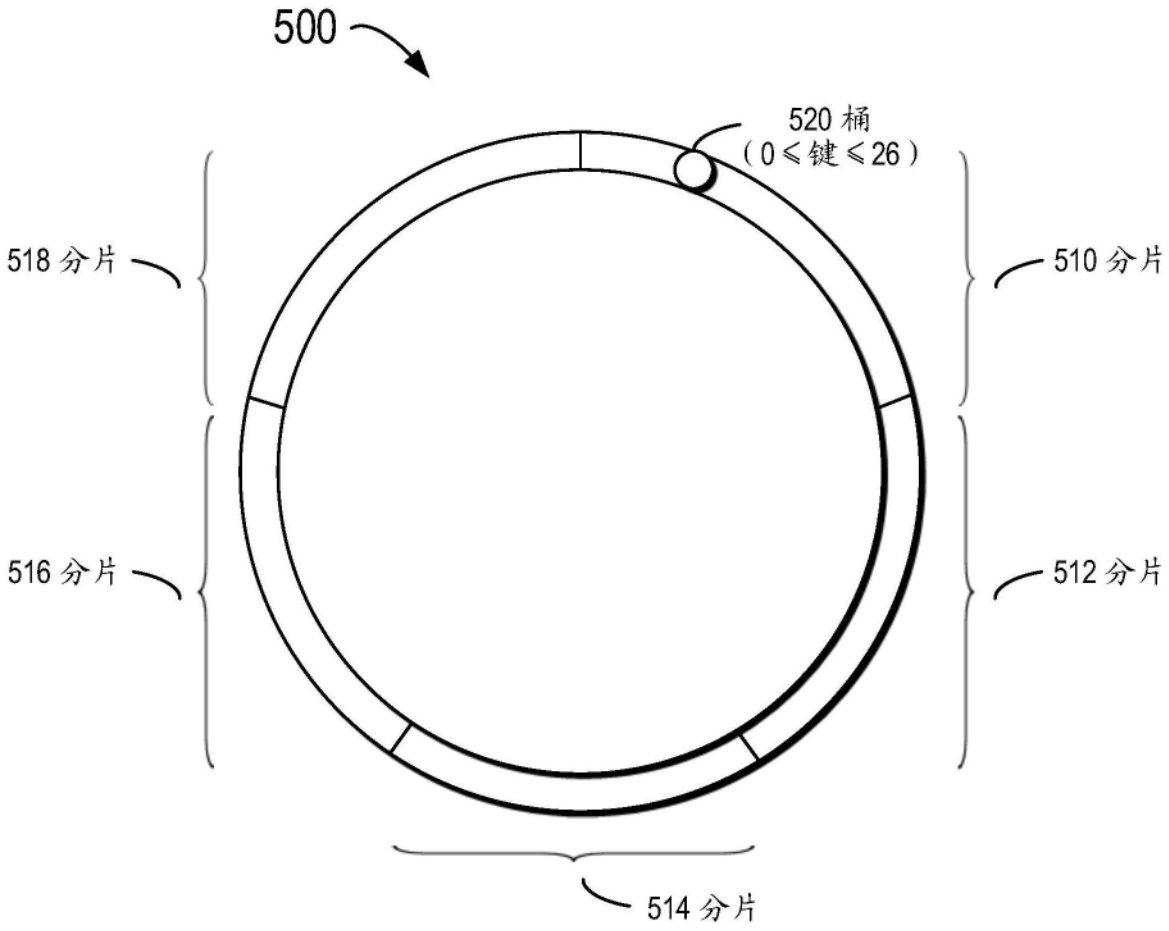


图5

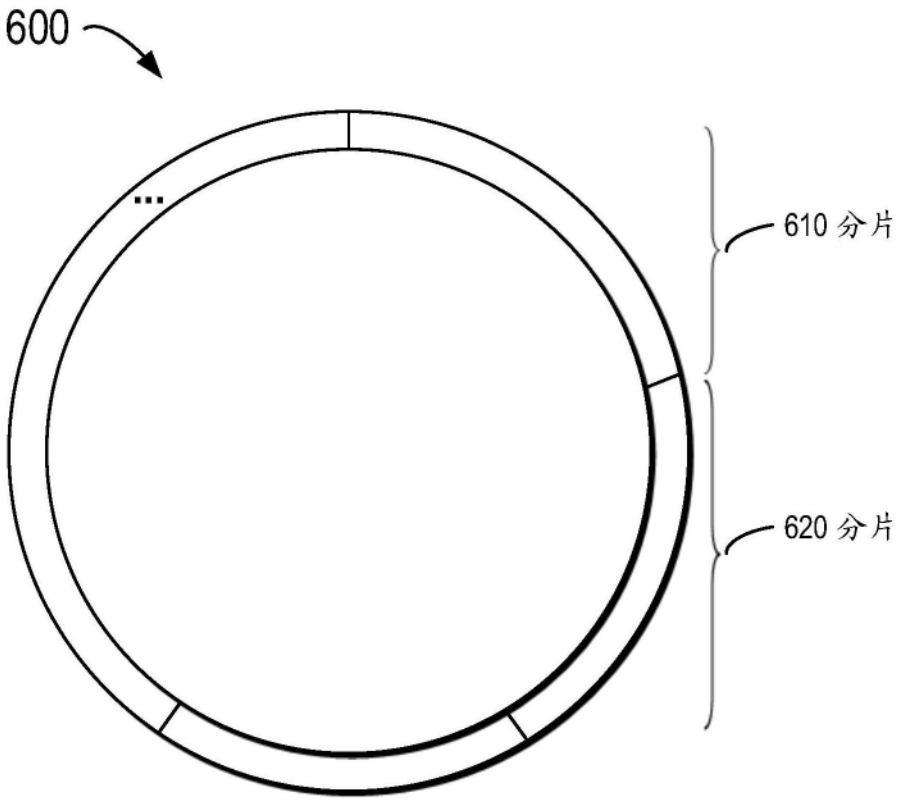


图6

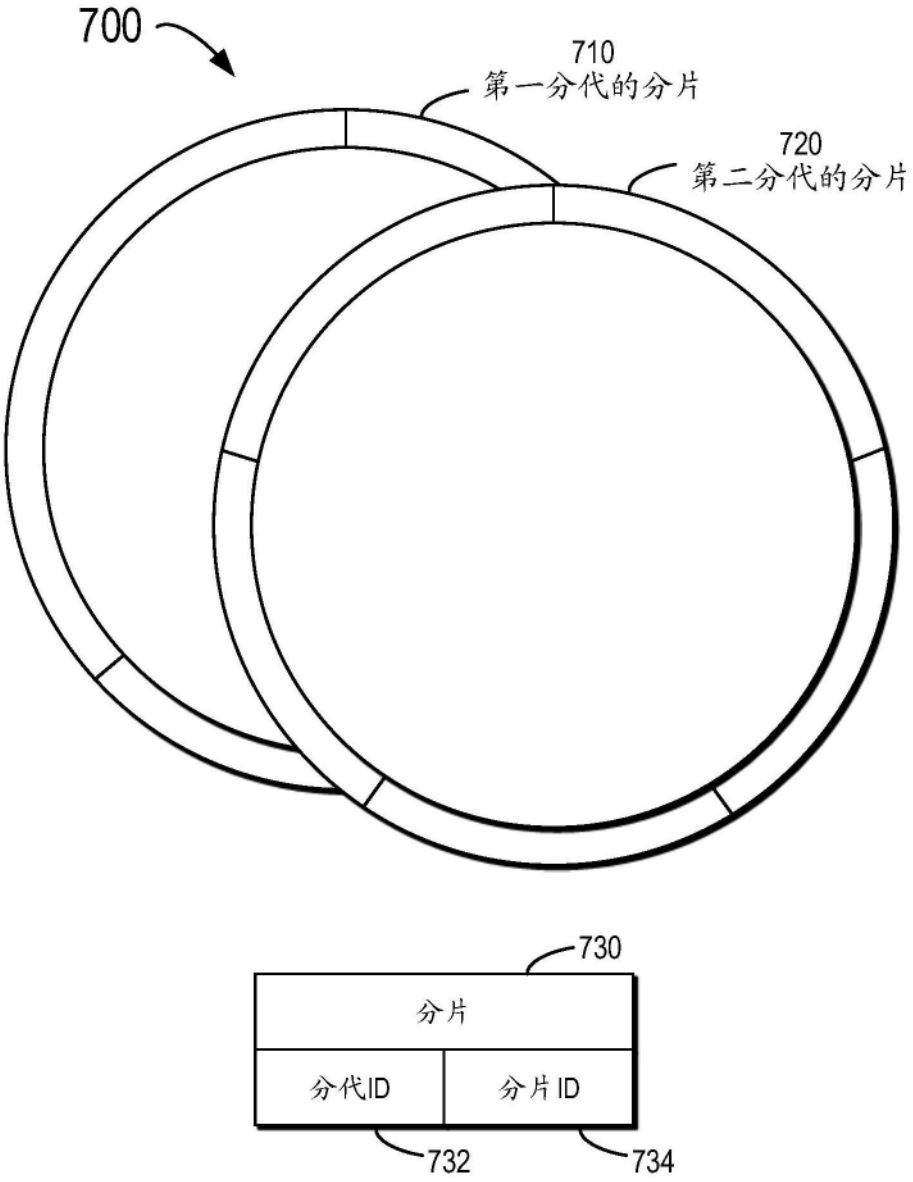


图7

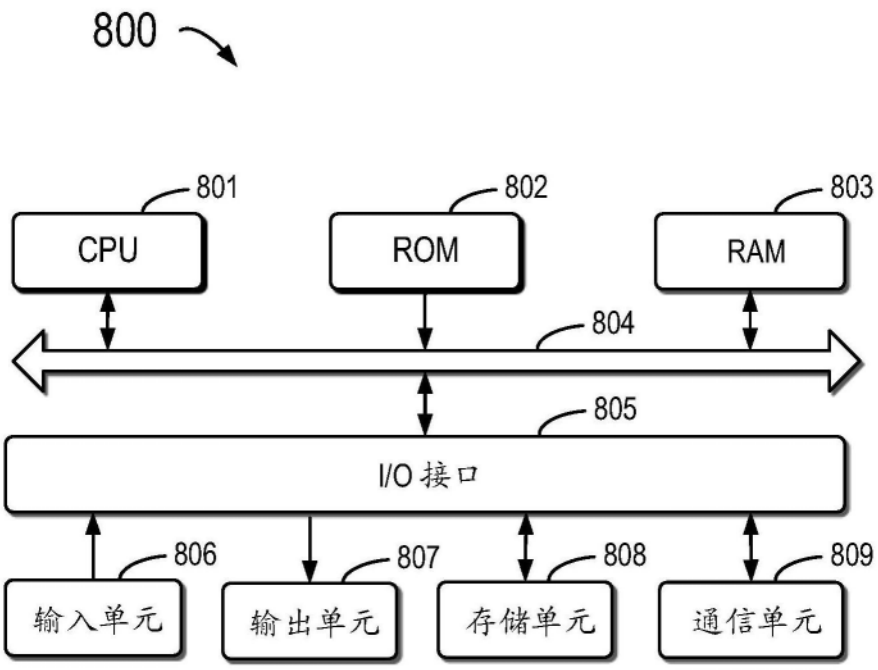

800
801
CPU
802
ROM
803
RAM
804
805
I/O 接口
806
输入单元
807
输出单元
808
存储单元
809
通信单元

图8