



(12) 发明专利

(10) 授权公告号 CN 103348314 B

(45) 授权公告日 2016. 01. 27

(21) 申请号 201180053028. 6

(22) 申请日 2011. 09. 14

(30) 优先权数据

12/882, 854 2010. 09. 15 US

(85) PCT国际申请进入国家阶段日

2013. 05. 03

(86) PCT国际申请的申请数据

PCT/US2011/051654 2011. 09. 14

(87) PCT国际申请的公布数据

W02012/037293 EN 2012. 03. 22

(73) 专利权人 净睿存储股份有限公司

地址 美国加利福尼亚

(72) 发明人 J·科尔格洛夫 J·海斯 洪波

王峰 E·米勒 C·哈莫

(74) 专利代理机构 中国国际贸易促进委员会专

利商标事务所 11038

代理人 叶勇

(51) Int. Cl.

G06F 3/06(2006. 01)

G06F 11/10(2006. 01)

G06F 12/02(2006. 01)

(56) 对比文件

US 2002065833 A1, 2002. 05. 30,

US 2002065833 A1, 2002. 05. 30,

US 2009249013 A1, 2009. 10. 01,

审查员 鱼冰

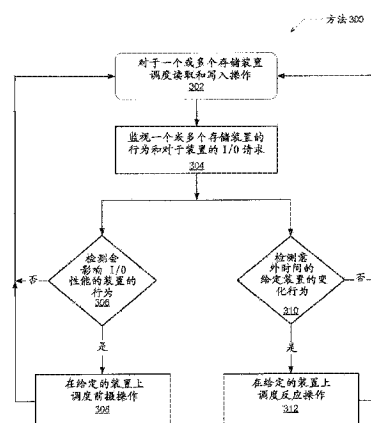
权利要求书2页 说明书15页 附图12页

(54) 发明名称

SSD 环境中的 I/O 的调度

(57) 摘要

提供用于有效地在多个固态存储装置之间调度读取和写入操作的系统和方法。计算机系统包括通过网络相互耦合的客户机计算机和数据存储阵列。数据存储阵列利用用于数据存储的固态驱动器和快擦写存储器单元。数据存储阵列内的存储控制器包含 I/O 调度器。为了在预测的时间使响应时间保持相对恒定,使用相应的存储装置的特性以调度对于存储装置的 I/O 请求。为了减少存储装置的未调度行为的可能性,存储控制器被配置为在存储装置上调度前摄操作,这将在减少未调度行为的出现的数量。



1. 一种计算机系统,包括:
包括被配置为存储数据的多个存储装置的数据存储介质;和
与数据存储介质耦合的数据存储控制器,
其中,数据存储控制器被配置为:
接收针对数据存储介质的读取和写入请求;
调度所述读取和写入请求以供所述多个存储装置处理;和
为了防止一个或多个存储装置表现包含可变性能的未调度行为,调度将由所述一个或多个存储装置执行的一个或多个前摄操作,其中,可变性能包含相对高响应等待时间或相对低吞吐量中的至少一个,
其中所述的调度一个或多个前摄操作包括调度高速缓存刷新操作,以防止给定的存储装置执行未调度的高速缓存刷新操作。
2. 如权利要求1所述的计算机系统,其中,存储控制器被配置为至少部分地基于 I/O 请求的最近历史调度所述一个或多个前摄操作。
3. 如权利要求1所述的计算机系统,其中,所述一个或多个前摄操作使得所述一个或多个存储装置中的至少一个进入表现相对高的响应等待时间的状态。
4. 如权利要求3所述的计算机系统,其中,所述多个存储装置是固态存储装置,并且其中,所述一个或多个前摄操作包含高速缓存刷新操作、安全擦除操作、修整操作、睡眠操作、休眠操作、电源接通和关断操作和复位操作中的一个或多个。
5. 如权利要求2所述的计算机系统,其中,由于未知原因,给定装置的未调度行为由该给定装置表现。
6. 如权利要求1所述的计算机系统,其中,所述一个或多个前摄操作被配置为将所述一个或多个存储装置中的每一个置于已知状态。
7. 如权利要求1所述的计算机系统,其中,存储装置包括至少一个 RAID 组,并且其中,存储控制器被配置为调度相对长等待时间操作,使得 RAID 组中的多个装置中的不多于 N 个的装置在任何给定的时间执行调度的长等待时间操作。
8. 如权利要求7所述的计算机系统,其中,当所述装置中的不多于 N 个在给定的时间繁忙时,那么可以执行重构读取,并且其中,当所述装置中的多于 N 个在给定的时间繁忙时,那么不能执行重构读取。
9. 如权利要求1所述的计算机系统,其中,所述数据存储控制器进一步被配置为:
检测所述多个存储装置中的给定存储装置表现未调度行为;和
响应于检测到出现所述给定存储装置的未调度行为,调度一个或多个反应操作,所述一个或多个反应操作被配置为使得所述给定存储装置进入已知状态。
10. 如权利要求1所述的计算机系统,其中,所述数据存储控制器被配置为:
监视多个存储装置中的给定存储装置的状态;和
响应于预测未调度行为可能不存在前摄操作的情况下出现,对于给定的存储装置调度前摄操作。
11. 如权利要求1所述的计算机系统,其中,所述数据存储控制器被配置为在随机的时间调度所述一个或多个前摄操作。
12. 如权利要求1所述的计算机系统,其中,所述数据存储控制器被配置为调度一个或

多个前摄操作,以响应于检测到长等待时间操作已被调度,在给定的存储装置上跟随长等待时间操作。

13. 一种用于计算系统中的方法,该方法包括:

接收针对数据存储介质的读取和写入请求,数据存储介质包含被配置为存储数据的多个存储装置;

调度所述读取和写入请求以供所述多个存储装置处理;和

为了防止一个或多个存储装置表现包含可变性能的未调度行为,调度将由所述一个或多个存储装置执行的一个或多个前摄操作,其中,可变性能包含相对高响应等待时间或相对低吞吐量中的至少一个,

其中所述的调度一个或多个前摄操作包括调度高速缓存刷新操作,以防止给定的存储装置执行未调度的高速缓存刷新操作。

14. 如权利要求 13 所述的方法,还包括至少部分地基于 I/O 请求的最近历史调度所述一个或多个前摄操作。

15. 如权利要求 13 所述的方法,其中,所述一个或多个前摄操作使得所述一个或多个存储装置中的至少一个进入其表现相对高的响应等待时间的状态。

16. 如权利要求 15 所述的方法,其中,所述多个存储装置是固态存储装置,并且其中,所述一个或多个前摄操作包含高速缓存刷新操作、安全擦除操作、修整操作、睡眠操作、电源接通和关断操作、休眠操作、和复位操作中的一个或多个。

17. 如权利要求 13 所述的方法,其中,存储装置的未调度行为由于未知原因而发生。

18. 如权利要求 13 所述的方法,其中,所述一个或多个前摄操作被配置为将所述一个或多个存储装置中的每一个置于已知状态。

19. 如权利要求 15 所述的方法,其中,存储装置包括至少一个 RAID 组,并且其中,所述方法还包括调度相对长等待时间操作,使得 RAID 组中的多个装置中的不多于 N 个的装置在任何给定的时间执行调度的长等待时间操作。

20. 如权利要求 19 所述的方法,其中,当所述装置中的不多于 N 个在给定的时间繁忙时,那么可以执行重构读取,并且其中,当所述装置中的多于 N 个在给定的时间繁忙时,那么不能执行重构读取。

21. 如权利要求 13 所述的方法,还包括:

检测所述多个存储装置中的给定存储装置表现未调度行为;和

响应于检测到出现所述给定存储装置的未调度行为,调度一个或多个反应操作,所述一个或多个反应操作被配置为使得所述给定存储装置进入已知状态。

22. 如权利要求 13 所述的方法,还包括:

监视多个存储装置中的给定存储装置的状态;和

响应于预测未调度行为可能不存在前摄操作的情况下出现,对于给定的存储装置调度前摄操作。

23. 如权利要求 13 所述的方法,其中,还包括在随机的时间调度所述一个或多个前摄操作。

SSD 环境中的 I/O 的调度

技术领域

[0001] 本发明涉及计算机网络,更特别地,涉及计算数据存储系统。

背景技术

[0002] 随着计算机存储器存储和数据带宽的增加,商业管理的数据的量和复杂性增加。诸如数据中心的大规模分布式存储系统一般运行许多商业操作。分布式存储系统可与通过一个或多个网络互连的大量的客户机计算机耦合。如果分布式存储系统的任意部分具有不良的性能或者变得不可用,那么公司操作会受损或者完全停止。这种分布式存储系统寻求数据可用性的高标准和高性能功能。

[0003] 在存储系统自身内部,除了提供如何执行操作的步骤以外,文件系统和存储装置级输入/输出(I/O)调度器一般确定读取和写入操作的次序。例如,与依次的读取和写入操作相比,非依次的读取和写入操作对于存储装置来说执行起来是成本更高的(例如,在时间和/或资源上)。因此,I/O 调度器可尝试减少非依次操作。另外,I/O 调度器可提供诸如饥饿防止、请求合并和过程间公正的其它功能。

[0004] 在存储装置之间,至少读取和写入响应时间可能明显不同。这种不同可能是技术自身的特征。因此,与选择的数据存储装置相关的技术和机构可确定用于执行有效的 I/O 调度的方法。例如,对于利用硬盘驱动(HDD)的系统,开发了许多当前的算法。HDD 包含一个或多个旋转的盘,每个盘涂有磁介质。这些盘以每分钟几千转的速度旋转。另外,电磁致动器负责在旋转的盘上定位磁读取/写入装置。装置的机械和电气机械设计影响其 I/O 特性。不幸的是,摩擦、磨损、振动和机械不对准会产生可靠性问题并且影响 HDD 的 I/O 特性。许多当前的 I/O 调度器被设计为考虑 HDD 的输入/输出(I/O)特性。

[0005] 另一种类型的存储介质的一个例子是固态驱动(SSD)。与 HDD 相对,SSD 利用固态存储器而不是磁介质装置以存储永久数据。固态存储器可包含快擦写存储器单元。快擦写存储器具有大量的与硬盘驱动器不同的特征。例如,快擦写存储器单元一般在被重写或重新编程之前在大的块中被擦除。快擦写存储器还一般在诸如裸片、封装、面和块的复杂配置中被组织。选择的配置的尺寸和平行性、快擦写存储器随时间的磨损和装置的互连和传送速度均会改变。另外,这些装置还会包括用于管理装置上的存储的快擦写翻译层(FTL)。由 FTL 利用的算法可改变,并且还会有助于装置的行为和/或性能的变化。因此,在对于存储使用基于快擦写的 SSD 同时利用对于诸如具有不同的特性的硬盘驱动器的系统设计的 I/O 调度器的系统中,一般不能实现高性能和可预测的等待时间。

[0006] 鉴于以上的情况,希望用于有效地调度多个存储装置之间的读取和写入操作的系统和方法。

发明内容

[0007] 公开了用于有效地在多个固态存储装置之间调度读取和写入操作的计算机系统和各种实施例。

[0008] 在一个实施例中,计算机系统包括被配置为在网络上向通过网络耦合为接收读取和写入请求的一个或多个数据存储阵列传送读取和写入请求的多个客户机计算机。设想在多个存储装置上包含多个存储位置的数据存储阵列。在各种实施例中,在用于数据存储和保护独立驱动配置的冗余阵列(RAID)中配置存储装置。数据存储装置可包含用于数据存储的固态存储器技术,诸如快擦写存储器单元。相应的存储装置的特性被用于调度对于存储装置的 I/O 请求。特性可包含对于 I/O 请求的预测响应时间、装置年龄、任何相应的高速缓存尺寸、存取速度、错误率、当前 I/O 请求和完成的 I/O 等。

[0009] 在一个实施例中,I/O 调度器被配置为接收读取和写入请求,并调度读取和写入请求以供多个存储装置处理。存储装置可根据服务的操作表现改变的等待时间,并且还会在各种时间表现未调度或不可预测的行为,这使得性能与期望或希望的情况不同。在各种实施例中,这些行为与装置适当地起作用(即,不处于错误状态)的行为对应,但是,基于等待时间和/或吞吐量在低于期望或希望的水平上执行。这种行为和性能可被称为“可变性能”行为。这些可变性能行为可例如由诸如基于快擦写的存储器技术的技术表现。为了减少装置的这种行为的出现,在一个实施例中,调度器被配置为在某些时间在一个或多个存储装置上调度前摄操作。前摄操作可一般减少装置表现未调度行为的可能性。在一些实施例中,调度器监视装置和系统的状态,并基于未调度行为即将到来或者将另外出现的预测对于给定的装置调度前摄操作。在一些实施例中,调度器在随机的时间对于存储装置调度前摄操作。

[0010] 考虑以下的描述和附图,这些和其它实施例将变得十分明显。

附图说明

[0011] 图 1 是示出网络结构的一个实施例的一般化框图。

[0012] 图 2 示出根据计算系统的一个实施例的概念模型。

[0013] 图 3 是示出用于调整 I/O 调度以减少数据存储子系统上的不可预测可变 I/O 响应时间的方法的一个实施例的一般化流程图。

[0014] 图 4 是示出用于将分发给存储装置的操作分开的方法的一个实施例的一般化框图。

[0015] 图 5 是示出用于发展用于表征存储子系统内的存储装置的行为的模型的方法的一个实施例的一般化流程图。

[0016] 图 6 是示出存储子系统的一个实施例的一般化框图。

[0017] 图 7 是示出装置单元的另一实施例的一般化框图。

[0018] 图 8 是示出状态表的另一实施例的一般化框图。

[0019] 图 9 是示出用于调整 I/O 调度以减少数据存储子系统上的不可预测可变 I/O 响应时间的方法的一个实施例的一般化流程图。

[0020] 图 10 是示出在共享的数据存储上以有效的等待时间保持读取操作的方法的一个实施例的一般化流程图。

[0021] 图 11 是示出用于减少表现可变 I/O 响应时间的存储装置的数量的一种方法的一个实施例的一般化流程图。

[0022] 图 12 是示出在共享的数据存储上以有效的等待时间保持读取操作的方法的一个

实施例的一般化流程图。

[0023] 虽然能够对于本发明提出各种修改和替代形式,但在附图中作为例子表示并且在这里详细描述特定的实施例。但应理解,其附图和详细的描述不是要将本发明限于特定的公开的形式,相反,本发明要覆盖落入由所附的权利要求限定的本发明的精神和范围内的所有的变更方案、等同方案和替代方案。

具体实施方式

[0024] 在以下的描述中,阐述大量的特定的细节以使得能够彻底理解本发明。但是,本领域技术人员可以想到,可以在没有这些特定的细节的情况下实现本发明。在一些情况下,为了避免混淆本发明,没有详细表示公知的电路、结构、信号、计算机程序指令和技术。

[0025] 参照图 1,表示网络结构 100 的一个实施例的一般化框图。如后面进一步描述的那样,网络结构 100 的一个实施例包含通过网络 180 相互连接并与数据存储阵列 120a ~ 120b 连接的客户机计算机系统 110a ~ 110b。网络 180 可通过交换机 140 与第二网络 190 耦合。客户机计算机系统 110c 通过网络 190 与客户机计算机系统 110a ~ 110b 和数据存储阵列 120a ~ 120b 耦合。另外,网络 190 可通过交换机 150 与因特网 160 或其它的外部网络耦合。

[0026] 注意,在替代性实施例中,客户机计算机和服务器、交换机、网络、数据存储阵列和数据存储装置的数量和类型不限于图 1 所示的那些。在各种时间,一个或多个客户机可离线操作。另外,在操作中,当用户连接、断开和重新连接网络结构 100 时,各单个客户机计算机连接类型可改变。并且,虽然本说明书一般讨论网络固定存储,但是,这里描述的系统和方法也可被应用于直接固定存储系统,并且可包括被配置为执行描述的方法的一个或多个方面的主机操作系统。大量的这种替代方案是可能的并被设想。简要提供图 1 所示的部件中的每一个的进一步的细节。首先,描述由数据存储阵列 120a ~ 120b 提供的特征中的一些的概要。

[0027] 在网络结构 100 中,数据存储阵列 120a ~ 120b 中的每一个可用于在诸如客户机计算机系统 110a ~ 110b 的不同的服务器和计算机之间共享数据。另外,数据存储阵列 120a ~ 120b 可被用于盘镜像、备份和恢复、存档和存档数据的检索和从一个存储装置到另一个的数据迁移。在替代性实施例中,为了形成群,一个或多个客户机计算机系统 110a ~ 110c 可通过快速局域网络(LAN)相互链接。这种客户机可共享诸如驻留于数据存储阵列 120a ~ 120b 中的一个内的群共享容量的存储资源。

[0028] 数据存储阵列 120a ~ 120b 中的每一个包含用于数据存储的存储子系统 170。存储子系统 170 可包含多个存储装置 176a ~ 176m。这些存储装置 176a ~ 176m 可向客户机计算机系统 110a ~ 110c 提供数据存储服务。存储装置 176a ~ 176m 中的每一个使用用于执行数据存储的特定的技术和机构。在存储装置 176a ~ 176m 中的每一个中使用的技术和机构的类型可至少部分地被用于确定用于控制和调度对于存储装置 176a ~ 176m 中的每一个的读取和写入操作的算法。在这些算法中使用的逻辑可包含于基本操作系统(OS)116、文件系统 140、存储子系统控制器 174 内的一个或多个全局 I/O 调度器 178、存储装置 176a ~ 176m 中的每一个内的控制逻辑中的一个或多个中,或者包含于其它位置。另外,这里描述的逻辑、算法和控制机构可包含硬件和 / 或软件。

[0029] 存储装置 176a ~ 176m 中的每一个可被配置为接收读取和写入请求,并包含多个数据存储位置,每个数据存储位置可寻址到阵列中的行和列。在一个实施例中,存储装置 176a ~ 176m 内的数据存储位置可被配置成逻辑、冗余存储容器或 RAID 阵列(便宜/独立盘的冗余阵列)。在一些实施例中,存储装置 176a ~ 176m 中的每一个可利用与常规的硬盘驱动(HDD)不同的数据存储的技术。例如,存储装置 176a ~ 176m 中的一个或多个可包含由固态存储器构成的存储装置或者进一步与其耦合以存储永久数据。在其它的实施例中,存储装置 176a ~ 176m 中的一个或多个可包含使用诸如旋转扭矩传输技术、磁阻随机存取存储器(MRAM)、叠瓦盘、忆阻器、相变存储器或其它存储技术的技术的存储器,或者进一步与其耦合。这些不同的存储技术可使得存储装置之间的不同的 I/O 特性。

[0030] 在一个实施例中,包含的固态存储器包含固态驱动(SSD)技术。一般地,SSD 技术利用快擦写存储器单元。在本领域中,众所周知,快擦写存储器单元基于在浮动门中捕获和存储的电子的范围保持二进制值。完全擦除快擦写存储器单元在浮动门中不存储或者存在最少量的电子。特定的二进制值,诸如用于单级单元(SLC)快擦写的二进制值 1,与擦除的快擦写存储器单元相关。多级单元(MLC)快擦写具有与擦除的快擦写存储器单元相关的二进制值 11。在向快擦写存储器单元内的控制门施加比给定的阈值电压高的电压之后,快擦写存储器单元在浮动门中捕获给定范围的电子。因此,另一特定的二进制值,诸如用于 SLC 快擦写的二进制 0,与编程(写入)快擦写存储器单元相关。MLC 快擦写单元可根据向控制门施加的电压具有与编程的存储器单元相关的多个二进制值中的一个。

[0031] HDD 技术和 SSD 技术之间的技术和机构的不同会使得数据存储装置 176a ~ 176m 的输入/输出(I/O)特性的不同。一般而言,SSD 技术提供的读取访问等待时间比 HDD 技术低。但是,SSD 的写入性能一般比读取性能慢,并且会明显受 SSD 内的自由可编程块的可用性影响。由于 SSD 的写入性能与 SSD 的读取性能相比明显较慢,因此,期望与读取类似的等待时间的某些功能或操作会出现问题。另外,由于影响读取等待时间的长写入等待时间,会使得调度更加困难。因此,对于数据存储阵列 120a ~ 120b 中的每一个中的 I/O 调度,可以使用不同的算法。

[0032] 在一个诸如读取和写入操作的不同类型的操作具有不同的等待时间的实施例中,用于 I/O 调度的算法会分开这些操作并且出于调度的目的单独地处理它们。例如,在存储装置 176a ~ 176m 中的一个或多个中,写入操作可通过装置自身被批处理,诸如通过将它们存储于内部高速缓存中。当这些高速缓存达到给定的占据阈值或者在一些其它的时间,相应的存储装置 176a ~ 176m 可刷新高速缓存。一般地,这些高速缓存刷新会在不可预测的时间对于读取和/或写入引入添加的等待时间,这使得有效地调度的操作变得困难。因此,为了预测什么时候会出现这种高速缓存刷新,I/O 调度器可利用存储装置的特性,诸如高速缓存的尺寸或测量的空闲时间。已知一个或多个存储装置 176a ~ 176m 中的每一个的特性可使得更有效的 I/O 调度。在一个实施例中,全局 I/O 调度器 178 可检测存储装置 176a ~ 176m 中的一个或多个的给定的装置在预测不到的时间对于 I/O 请求表现长的响应时间。作为响应,为了使得装置重新开始展示期望的行为,全局 I/O 调度器 178 可对于给定的装置调度给定的操作。在一个实施例中,这种操作可以是高速缓存刷新命令、修整命令或擦除命令等。将讨论在后面关于 I/O 调度的其它细节。

[0033] 网络构架的部件

[0034] 并且,如图所示,网络构架 100 包含通过网络 180 和 190 互连并与数据存储阵列 120a ~ 120b 连接的客户机计算机系统 110a ~ 110c。网络 180 和 190 可包括包含无线连接、直接局域网(LAN)连接、广域网(WAN)连接的各种技术,诸如因特网、路由器、存储区域网络和以太网等。网络 180 和 190 可包含一个或多个 LAN,这些 LAN 也可以是无线的。网络 180 和 190 还可包含远程直接存储器存取(RDMA)硬件和 / 或软件、传输控制协议 / 内部协议(TCP/IP)硬件和 / 或软件、路由器、重复器、交换机和 / 或网格等。可以在网络 180 和 190 中使用诸如光纤通道、以太网光纤通道(FCoE)和 iSCSI 等的协议。交换机 140 可利用与网络 180 和 190 相关的协议。网络 190 可与用于因特网 160 的诸如传送控制协议(TCP)和因特网协议(IP)或 TCP/IP 的一组通信协议连接。交换机 150 可以是 TCP/IP 交换机。

[0035] 客户机计算机系统 110a ~ 110c 可代表任意数量的静止或移动计算机,诸如桌面个人计算机(PC)、服务器、服务器农场、工作站、膝上型计算机、手持计算机、服务器、个人数字助理(PDA)和智能电话等。一般而言,客户机计算机系统 110a ~ 110b 包括包含一个或多个处理器芯的一个或多个处理器。各处理器芯包含用于根据预定的通用指令集执行指令的电路。例如,可以选择 x86 指令集构架。作为替代方案,可以选择 **Alpha®**、**PowerPC®**、**SPARC®** 或任何其它通用指令集构架。处理器芯可对于数据和计算机程序指令访问高速缓存存储器子系统。高速缓存子系统可与包含随机存取存储器(RAM)和存储装置的存储器层次耦合。

[0036] 客户机计算机系统内的各处理器芯和存储器层次可与网络接口连接。除了硬件部件以外,客户机计算机系统 110a ~ 110c 中的每一个可包含存储于计算机层次内的基本操作系统(OS)。基本 OS 可代表各种操作系统中的任一种,诸如,例如, **MS-DOS®**、**MS-WINDOWS®**、**OS/2®**、**UNIX®**、**Linux®**、**Solaris®**、**AIX®** 或 DART 等。因而,基本 OS 可操作为向最终用户提供各种服务,并提供可操作为支持各种程序的执行的软件框架。另外,客户机计算机系统 110a ~ 110c 中的每一个可包含用于支持虚拟机(VM)的管理程序。本领域技术人员公知,可在桌面和服务器的使用虚拟化,以完全或部分地从系统的硬件将诸如 OS 的软件去耦合。虚拟化可向最终用户提供多个 OS 在同一机器上运行的假象,每个 OS 具有其自身的资源和对于在数据存储阵列 120a ~ 120b 中的每一个内的存储装置 176a ~ 176m 上构建的逻辑存储实体(例如, LUN) 的访问。

[0037] 数据存储阵列 120a ~ 120b 中的每一个可被用于在诸如客户机计算机系统 110a ~ 110c 的不同的服务器之间共享数据。数据存储阵列 120a ~ 120b 中的每一个包含用于数据存储的存储子系统 170。存储子系统 170 可包含多个存储装置 176a ~ 176m。这些存储装置 176a ~ 176m 中的每一个可以是 SSD。控制器 174 可包含用于操作接收的读取 / 写入请求的逻辑。例如,可至少在控制器 174 中执行以上简要描述的算法。随机存取存储器(RAM) 712 可被用于批处理操作,诸如接收的写入请求。在各种实施例中,当批处理写入操作(或其它操作)时,可以使用非易失性存储(例如, NVRAM)。

[0038] 基本 OS132、文件系统 134、任何 OS 驱动程序(未示出)和存储于存储器介质 130 中的其它软件可提供访问文件的功能和这些功能的管理。基本 OS134 和 OS 驱动程序可包含存储于计算机介质 130 上并且可通过处理器 122 执行以在与接收的请求对应中的存储子系

统 170 中执行一个或多个存储器访问操作的程序指令。图 1 所示的系统可一般包含一个或多个文件服务器和 / 或块服务器。

[0039] 数据存储阵列 120a ~ 120b 中的每一个可使用网络接口 124 以与网络 180 连接。与客户机计算机系统 110a ~ 110c 类似, 在一个实施例中, 可在网络适配器卡上包含网络接口 124 的功能。可通过使用硬件和软件实现网络接口 124 的功能。可在网络接口 124 的网络卡实现上包含随机存取存储器 (RAM) 和只读存储器 (ROM)。可以使用一个或多个应用特定集成电路 (ASIC) 以提供网络接口 124 的功能。

[0040] 在一个实施例中, 可以开发寻求最优化 I/O 性能的数据存储模型。在一个实施例中, 模型至少部分地基于存储系统内的存储装置的特性。例如, 在利用固态存储技术的存储系统中, 可以利用特定的装置的特性以开发用于装置的模型, 这些模型又用于通知相应的 I/O 调度算法。例如, 如果使用的特定的存储装置表现与读取等待时间相比相对高的写入等待时间, 那么这种特性可应对调度操作。注意, 所视为相对高或低的可根据给定的系统、被处理的数据的类型、处理的数据的量或数据的定时等改变。一般而言, 系统可被编程, 以确定什么构成低等待时间或高等待时间, 以及 / 或者什么构成两者之间的明显的不同。

[0041] 一般而言, 对于装置或计算系统开发的任何模型将是不完整的。在真实世界系统中经常存在太多的需要应对的变量以完全将给定的系统模型化。在一些情况下, 可能能够开发不完整但仍然有价值的模型。如下面更完整地讨论的那样, 描述基于装置的特性将存储装置模型化的实施例。在各种实施例中, 关于装置会如何表现基于某些预测执行 I/O 调度。基于装置的特性的理解, 某些装置行为比其它行为更容易预测。为了对于最佳的 I/O 性能更有效地调度操作, 希望更多地控制系统的行为。意外或者不可预测的装置行为使得调度操作更困难。因此, 开发寻求使系统中的不可预测或意外的行为最小化的算法。

[0042] 图 2 提供被模型化的装置或系统和用于将装置或系统内的不可预测的行为最小化方法的概念图。在第一块 200 中, 示出理想方案。在块 200 中表示系统 204 和系统的模型 202。在一个实施例中, 系统可以是单个装置。作为替代方案, 系统可包含许多装置和 / 或部件。如上面讨论的那样, 模型 202 可能不是寻求模型化的系统 204 的完整模型。然而, 出于模型的目的, 模型 202 捕获关注的行为。在一个实施例中, 模型 202 可寻求将计算存储系统模型化。在理想方案 200 中, 系统 204 的实际行为与模型 202 的行为“一致”。换句话说, 系统 204 的行为一般表现模型 202 寻求捕获的那些行为。当系统行为 204 根据模型 202 的行为时, 系统行为一般会更容易预测。因此, 可更有效地执行系统内的操作 (例如, 读取和写入操作) 的调度。

[0043] 例如, 如果希望最优化读取响应时间, 那么可能能够调度读取, 使得, 如果系统的其它行为是相对可预测的, 那么它们以更加适时的方式服务。另一方面, 如果系统行为相对不可预测, 那么调度这些读取以在希望时提供结果的能力的置信度降低。块 210 示出系统行为 (较小的圆) 不与系统的模型 (较大圆) 对准的方案。在这种情况下, 系统表现落于模型外面的行为。因此, 系统行为是不可预测的, 并且操作的调度会变得低效。例如, 如果在存储系统中使用固态存储器装置并且这些装置可在它们自身上启动使得装置以更大 (或另外意外) 等待时间上服务请求的动作, 那么对于该装置调度的任何操作也可经历更大或意外的等待时间。这种装置操作的一个例子是内部高速缓存刷新。

[0044] 为了解决意外或未调度系统行为和相应的可变性能的问题, 开发的模型可包含可

采取以将系统恢复到低不确信状态的动作。换句话说,假如系统开始表现使模型预测系统行为的能力劣化的行为,则模型已在其中内置了它可采取的将系统恢复到消除或者不可能呈现特定的意外行为的状态的某些动作。在表示的例子中,表示寻求将系统“移动”到更紧密地与模型一致的状态的动作 212。动作 212 可被称为“反应”动作或操作,原因是它是响应检测处于模型外面的系统行为被执行的。在执行动作 212 之后,可以实现更理想的状态 220。

[0045] 虽然希望开发可对意外行为起反应以将系统移动到更理想的状态的模型,但是,这些意外的行为的存在仍会干涉有效的调度操作。因此,希望使意外行为或事件的出现最小化。在一个实施例中,开发包含被设计为防止或减少意外行为的出现的动作或操作的模型。这些动作可被称为“前摄”动作或操作,原因是,为了防止出现一些行为或事件或者改变一些行为或事件的定时,它一般可被前摄地执行。图 2 中的块 230 示出系统行为(较小的圆)处于模型(较大的圆)内的方案。然而,模型采取动作 232 以移动系统行为以使得它保持处于模型内并且可能更理想地一致。块 230 中的系统行为可被视为接近它表现模型外面的行为的状态。在这种情况下,模型可具有用于相信系统接近这种状态的一些基础。例如,如果 I/O 调度器已向给定的装置传送大量的写入操作,那么调度器可期望装置可在将来的一些时间执行内部高速缓存刷新操作。调度器不是等待这种事件的出现,而是可前摄地对于该装置调度高速缓存刷新操作,使得在调度器的选择时间执行高速缓存刷新。作为替代方案,或者,除了以上情况以外,可在随机的时间执行这种前摄操作。虽然仍出现高速缓存刷新,但是,其出现不是意外的,并且它现在变为由调度器执行的总调度的一部分,并且可以以更有效和智能的方式被管理。在执行该前摄动作 232 之后,系统一般可被视为处于更容易预测的状态 240。这是由于在装置上调度和执行了高速缓存刷新,并且,装置自发地启动内部高速缓存刷新的可能性降低(即,其高速缓存已被刷新)。通过组合模型内的反应和前摄动作或操作,可实现更大的系统可预测性,并且可实现改进的调度。

[0046] 现在参照图 3,该图示出用于执行 I/O 调度以减少意外行为的方法 300 的一个实施例。在上述的网络构架 100 和数据存储阵列 120a ~ 120b 中体现的部件可一般根据方法 300 操作。本实施例中的步骤以依次的方式被表示。但是,一些步骤可以以与所示的次序不同的次序出现,一些步骤可同时被执行,一些步骤可与其它的步骤组合,并且,在另一实施例中可缺少一些步骤。

[0047] 在块 302 中, I/O 调度器对于一个或多个存储装置调度读取和写入操作。在各种实施例中, I/O 调度器可对于各存储装置保持单独的队列(在物理上或逻辑上)。另外, I/O 调度器可包含用于由相应的存储装置支持的各操作类型的单独的队列。例如, I/O 调度器可对于 SSD 至少保持单独的读取队列和单独的写入队列。在块 304 中, I/O 调度器可监视一个或多个存储装置的行为。在一个实施例中, I/O 调度器可包含相应的存储装置的模型(例如,行为类型模型和 / 或至少部分地基于装置的模型的算法),并且从存储装置接收用于输入到模型的状态数据。I/O 调度器内的模型可通过使用存储装置的已知和 / 或观察的特性模型化和预测存储装置的行为。

[0048] I/O 调度器可检测给定的存储装置的影响或者可能影响 I/O 性能的特性。例如,如后面讨论的那样,可以保持装置和 I/O 流量的各种特性和状态。通过观察这些特性和状态, I/O 调度器可预测给定的装置可能很快进入它表现高 I/O 等待时间行为的状态。例如,

在一个实施例中, I/O 调度器可检测或预测要在存储装置内出现会影响对于存储装置的请求的响应时间的内部高速缓存刷新。例如, 在一个实施例中, 对于给定的时间量处于空闲状态的存储装置会刷新其内部高速缓存。在一些实施例中, 给定的装置是否空闲可基于装置外部的观点。例如, 如果对于某时间周期对于装置还没有调度操作, 那么装置可被视为空闲大约预定的时间周期。在这种实施例中, 基于装置内的内部启动的活动, 装置可能事实上是繁忙的。但是, 在确定装置是否空闲时, 这种内部启动的活动不会被考虑。在其它的实施例中, 当确定装置是空闲还是繁忙时, 可以考虑装置的内部启动的活动。通过观察装置的行为并关注它已空闲给定的时间, 调度器可预测什么时候会出现内部高速缓存刷新。在其它的实施例中, 调度器还可具有轮询装置以确定装置的各种状态或条件的能力。在任意的情况下, 调度器可被配置为确定诸如内部高速缓存刷新的未调度行为的可能性, 并且, 为了防止出现该行为, 启用前摄操作。以这种方式, 调度器控制装置和系统中的事件的定时, 并且能够更好地调度操作。

[0049] 作为关于装置行为进行预测的基础, 可以使用各种实施例。在各种实施例中, 调度器可保持当前未决的操作的状态和 / 或与存储装置对应的最近的操作的历史。在一些实施例中, I/O 调度器可获知装置内的高速缓存的尺寸和 / 或高速缓存策略, 并且保持发送到存储装置的写入请求的数量的计数。在其它的实施例中, 其它的机构可用于确定装置内的高速缓存的状态(例如, 直接轮询对于装置的类型访问)。另外, I/O 调度器可跟踪发送到存储装置的写入请求的数据的量。I/O 调度器可然后检测写入请求的数量或与写入请求对应的数据的总量什么时候达到给定的阈值。如果 I/O 调度器检测到这种条件(条件块 306), 那么在块 308 中, I/O 调度器可调度装置的特定的操作。这种操作一般可与上述的前摄操作对应。例如, I/O 调度器可在相应的队列中放置高速缓存刷新命令, 以强制存储装置在调度器的选择时间执行高速缓存刷新。作为替代方案, 为了确定是否完成存储装置上的任何高速缓存刷新, I/O 调度器可在队列中放置伪读取操作。并且, 调度器可询问装置以获得状态信息(例如, 空闲、繁忙等)。这些和其它的特性和操作是可能的, 并且被设想。另外, 在各种实施例中, 前摄操作可在将 SSD 重调于适当的位置时被调度。在这种实施例中, SSD 固件和 / 或映射表可进入请求延缓或者永久较慢的状态。可能能够仅将驱动复位或者关断和接通驱动以使固件畅通。但是, 如果条件是永久(即, 不能处理映射表的当前状态的固件中的错误), 另一种固定它的方式是, 重新格式化驱动以完全清除和复位 FTL, 并然后重新占据它或者重新使用它以用于一些其它的数据。

[0050] 可以执行以上描述的动作以防止或减少意外可变响应时间的出现的次数。同时, I/O 调度器可检测意外时间上的给定存储装置的任何可变行为的出现。如果 I/O 调度器检测到这种条件(条件块 310), 那么, 在块 312 中, I/O 调度器可将操作置于存储装置的相应的队列中。在这种情况下, 操作一般可与上述的反应操作对应。可以使用该操作以减少存储装置提供可变行为并检测可变行为的结束的时间。在各种实施例中, 前摄和 / 或反应操作一般可包含能够将装置置于(至少部分)已知状态的任何操作。例如, 启动高速缓存刷新操作可使得装置实现空的高速缓存状态。与高速缓存不为空的装置相比, 具有空的高速缓存的装置更不可能启动内部高速缓存刷新。前摄和 / 或反应操作的一些例子包括高速缓存刷新操作、擦除操作、安全擦除操作、修整操作、睡眠操作、休眠操作、电源接通和关断操作和复位操作。

[0051] 现在参照图 4, 该图示出用于分开分给存储装置的操作的方法 400 的一个实施例。本实施例中的步骤以依次的方式被表示。但是, 一些步骤可以以与所示的次序不同的次序出现, 一些步骤可同时被执行, 一些步骤可与其它步骤组合, 并且, 在另一实施例中可缺少一些步骤。在各种实施例中, 出于调度的目的, 第一类型的操作可与第二类型的操作分开。例如, 在一个实施例中, 第一类型的操作可被赋予高于第二类型的操作的调度优先级。在这种实施例中, 第一类型的操作可被调度以进行相对迅速的处理, 而第二类型的操作排队以在后面处理(事实上, 延后操作的处理)。在给定的时间点上, 第一类型的操作的处理可暂停, 而(第二类型)的先前排队的操作被处理。随后, 第二操作类型的处理可重新停止, 而处理优先级返回到第一类型的操作。什么时候处理对一种类型暂停并且对于另一种类型开始可基于时间周期、累积数据、交易频率、可用资源(例如, 队列利用)、以上的任意的组合, 或者基于任何希望的条件。

[0052] 对于随机读取和写入请求, SSD 一般表现比 HDD 好的性能。但是, 由于 SSD 的特性, 与读取请求相比, SSD 一般对于随机写入请求表现较差的性能。与 HDD 不同, 读取和写入请求的相对等待时间相当不同, 使得写入请求花费的时间一般明显比读取请求长, 原因是, 与读取相比, 它占用更长的时间以将快擦写存储器单元编程。另外, 由于作为写入的一部分需要执行的附加的操作, 写入操作的等待时间的变化可相当大。例如, 可在已修改的快擦写存储器单元的写入或程序操作之前执行擦除操作。另外, 可在块的基础上执行擦除操作。在这种情况下, 处于块(擦除段)中的所有的快擦写存储器单元被一起擦除。由于块相对大并包含多个页, 因此, 操作可花费相对长的时间。作为替代方案, FTL 可将块重新映射到已擦除的擦除块中。在任意的情况下, 与读取相比, 与执行写入操作相关的附加的操作可使得写入具有明显较高的等待时间的可变性以及明显较高的等待时间。其它的存储装置类型可基于请求类型表现不同的特性。除了以上以外, 如果读取和写入请求被混合, 那么某些存储装置可能提供较差和 / 或可变的性能。因此, 为了提高性能, 各种实施例可分开读取和写入。注意, 虽然讨论一般特别提到读取和写入操作, 但是, 这里描述的系统和方法也可被应用于其它的操作。在这些其它的实施例中, 其它相对高和较低等待时间操作可因而被识别并出于调度的目的被分开。另外, 在一些实施例中, 读取和写入可被归为第一类型的操作, 而其它的操作, 诸如高速缓存刷新和修整操作, 可被归为与第二类型的操作对应。各种组合是可能的, 并且被设想。

[0053] 在块 402 中, I/O 调度器可接收并缓冲对于一个或多个存储装置的给定存储装置的 I/O 请求。在块 404 中, 低等待时间 I/O 请求一般会优先于高等待时间请求被发给存储装置。例如, 根据由存储装置使用的存储技术, 读取请求可具有比写入请求和其它命令类型低的等待时间, 并且, 可首先发出。因此, 写入请求可被累积, 而读取请求被赋予发出优先级(即, 在写入请求之前被传输给装置)。在一些时间点上, I/O 调度器可停止向装置发出读取请求并开始发出写入请求。在一个实施例中, 写入请求可作为多个写入的流被发出。因此, 与写入请求相关的费用可在多个写入请求上被分期偿还。以这种方式, 高等待时间请求(例如, 写入请求)和低等待时间请求(例如, 读取请求)可被分开并被单独地处理。

[0054] 在块 406 中, I/O 调度器可确定是否存在表示应向装置传输高等待时间请求的特定的条件。例如, 在一个实施例中, 检测这种条件可包含检测高等待时间 I/O 请求的给定的数量或相应的数据的量被累积并达到给定的阈值。作为替代方案, 被接收的高等待时间请

求的比率可达到一定的阈值。大量的这种条件是可能的并被设想。在一个实施例中,高等待时间请求可以是写入请求。如果出现这种条件(条件块 408),那么在块 410 中, I/O 调度器可开始向给定的存储装置发出高等待时间 I/O 请求。发出的这种请求的数量可根据给定的算法改变。数量可与固定或可编程的写入的数量或数据量对应。作为替代方案,可对于给定的时间发出写入。例如,时间周期可持续到特定条件停止存在(例如,接收的写入的比率下降)或者出现特定的条件。作为替代方案,在确定什么时候开始以及什么时候停止向装置发出高等待时间请求时,可以使用以上中的任一个的组合。在一些实施例中,与其它的读取请求相比,写入请求的流之后的第一读取请求可相对慢。为了避免调度在紧接着写入请求的流之后的发出时隙中调度“真正”读取请求, I/O 调度器可被配置为自动调度跟随写入请求的流的“伪”读取。在本文中,“真正”读取是通过用户或应用请求数据的读取,并且,“伪”读取是人为创建的可简单地舍弃数据的读取。在各种实施例中,在伪读取被检测为完成时,写入请求不能被确定为完成。并且,在各种实施例中,高速缓存刷新可跟随写入的流,并被用于确定什么时候完成写入。

[0055] 现在参照图 5,该图表示用于发展模型以表征存储子系统存储装置的行为的方法 500 的一个实施例。本实施例中的步骤以依次的方式被表示。但是,一些步骤可以以与所示的次序不同的次序出现,一些步骤可同时被执行,一些步骤可与其它步骤组合,并且,在另一实施例中可缺少一些步骤。

[0056] 在块 502 中,一个或多个存储装置可被选择以被用于存储子系统中。在块 54 中,可以识别各装置的各种特性,诸如高速缓存尺寸、典型的读取和写入响应时间、存储布局和装置的年龄等。在块 506 中,可以识别影响给定的存储装置的 I/O 性能的一个或多个特性。

[0057] 在块 508 中,可以确定影响给定装置的特性的定时和/或出现的一个或多个动作。例子可包括高速缓存刷新和诸如 SSD 的擦除操作的给定操作的执行。例如,诸如高速缓存刷新的强制操作可减少意外时间的 SSD 的可变响应时间的出现。在块 510 中,可基于相应的特性和动作对于一个或多个选择的装置中的每一个发展模型。可在软件中,诸如在存储控制器内的 I/O 调度器内,使用该模型。

[0058] 现在转到图 6,该图表示存储子系统的一个实施例的一般化框图。在表示的实施例中,存储装置 176a ~ 176m 中的每一个被示为处于单个装置组内。但是,在其它的实施例中,一个或多个存储装置 176a ~ 176m 可在装置组 173a ~ 173m 中的两个或更多个中被分割。在装置单元 600a ~ 600w 内可包括各存储装置的一个或多个相应的操作队列和状态表。这些装置单元可存储于 RAM172 中。对于装置组 173a ~ 173m 中的每一个,可以包括相应的 I/O 调度器 178。各 I/O 调度器 178 可包括跟踪相应的装置组内的存储装置中的每一个的状态数据的监视器 610。调度逻辑 620 可执行向相应的存储装置发出哪个请求的确定并确定发出请求的定时。

[0059] 现在转到图 7,该图示出装置单元 600 的一个实施例的一般化框图。装置单元 600 可包括装置队列 710 和表 720。装置队列 710 可包含读取队列 712、写入队列 714 和诸如其它的操作队列 716 的一个或多个其它的队列。各队列可包含用于存储一个或多个相应的请求的多个条目 730。例如,用于相应的 SSD 的装置单元可包含至少用于存储读取请求、写入请求、修整请求和擦除请求等的队列。表 720 可包含一个或多个状态表 722a ~ 722b,每个状态表包含用于存储状态数据的多个条目 730。在各种实施例中,图 7 所示的队列可以是物

理和 / 或逻辑分开的。还应注意,虽然队列和表被示为包含特定数量的条目,但是,条目自身未必相互对应。另外,队列和表的数量可与图所示的情况不同。另外,给定队列或队列间的条目可被区分优先次序。例如,读取请求可具有影响向装置发出请求的次序的高、中或低优先级。另外,可根据各种条件改变这种优先级。例如,达到一定年龄的低优先级读取可使其优先级增加。大量的这种优先方案和技术对于本领域技术人员来说是已知的。所有这些方法被设想,并且与这里描述的系统和方法相关地被使用。

[0060] 现在参照图 8,该图表示出诸如图 7 所示的状态表的一个实施例的一般化框图。在一个实施例中,这种表可包含与给定存储装置的状态、错误、损耗水平信息和其它信息对应的数据。相应的 I/O 调度器能够访问该信息,这可允许 I/O 调度器更好地调度对于存储装置的 I/O 请求。在一个实施例中,信息可至少包含装置年龄 802、错误率 804、在装置上检测的错误的总数 806、可恢复错误的数量 808、不可恢复错误的数量 810、装置的访问率 812、存储数据的年龄 814、相应的高速缓存尺寸 816、相应的高速缓存刷新空闲时间 818、用于分配空间的一个或多个分配状态 820 ~ 822、并发水平 824 和各种操作的期望时间 826 中的一个或多个。分配状态可包含填充、空置和错误等。给定装置的并发水平可包含关于装置同时处理多个操作的能力的信息。例如,如果装置具有 4 个快擦写芯片并且每一个能够一次完成一个传送,那么装置能够进行最多 4 个并行的操作。是否可并行执行特定的操作可依赖于数据如何在装置上展开。例如,如果装置内的数据在被请求访问的数据均处于一个芯片的位置上被展开,那么该数据上的操作可随访问不同的芯片上数据的请求并行地进行。但是,如果被请求访问的数据跨着多个芯片被剥夺,那么请求可相互干涉。因此,装置能够进行最多 N 个并行 / 并发操作(例如,在以上描述的装置具有 4 个芯片的情况下,为 4)。作为替代方案,最大并发水平可基于包含的操作的类型。在任意的情况下,调度器可在调度操作中考虑指示并发水平 N 和未决交易数量 M 的存储信息。

[0061] 现在参照图 9,该图表示用于调整 I/O 调度以减少数据存储子系统上的意外可变 I/O 响应时间的方法 900 的另一实施例。在上述的网络构架 100 和数据存储阵列 120a ~ 120b 中体现的部件可一般根据方法 900 操作。出于讨论的目的,本实施例中的步骤以依次的方式被表示。但是,一些步骤可以以与所示的次序不同的次序出现,一些步骤可同时被执行,一些步骤可与其它步骤组合,并且,在另一实施例中可缺少一些步骤。

[0062] 在块 902 中, I/O 调度器可监视存储装置中的每一个的行为。条件块 904 ~ 908 示出以上关于方法 300 的条件步骤 306 描述的会影响 I/O 性能的给定装置的检测特性的一个实施例。在一个实施例中,如果 I/O 调度器检测到给定的装置超过给定的空闲时间(条件块 904)或者检测到相应的高速缓存超过占据阈值(条件块 906)或者检测到高速缓存数据超过数据年龄阈值(条件块 908),那么,在块 910 中, I/O 调度器可向给定的存储装置发出强制(前摄)操作。在这种情况下,调度器可预测内部高速缓存刷新将很快在不可预测的时间出现。为了避免出现这种事件, I/O 调度器可前摄地调度操作以避免事件。

[0063] 注意,上述的事件的避免可意味着事件不出现或者不在不可预测或意外的时间出现。换句话说,调度器一般优选给定的事件根据调度器的定时而不是以其它的方式出现。在这种意义上,由于调度器调度事件出现的长等待时间事件比这种意外出现的事件好。调度逻辑 620 内的定时器和计数器可与监视器 160 组合使用,以至少执行这些检测。向给定的存储装置发出的强制操作的一个例子可包括高速缓存刷新。强制操作的另一例子可包括擦

除请求。强制操作可从 I/O 调度器可作为调度的一部分被发送到相应的装置单元 600 内的装置队列 710 中的相应的队列。

[0064] 现在参照图 10, 该图表示在共享的数据存储上以相对低的等待时间保持读取操作的方法 1000 的一个实施例。在上述的网络构架 100 和数据存储阵列 120a ~ 120b 中体现的部件可一般根据方法 1000 操作。出于讨论的目的, 本实施例中的步骤以依次的方式被表示。但是, 一些步骤可以以与所示的次序不同的次序出现, 一些步骤可同时被执行, 一些步骤可与其它步骤组合, 并且, 在另一实施例中可缺少一些步骤。

[0065] 在块 1002 中, 存储子系统 RAID 构架中的冗余量可被确定以被用于给定的装置组 173 中。例如, 对于 4+2RAID 组, 可以使用存储装置中的 2 个以存储诸如奇偶信息的擦除校正代码(ECC)信息。该信息可作为重构读取请求的一部分被使用。在一个实施例中, 可在正常 I/O 调度中使用重构的读取请求, 以提高装置组的性能, 同时, 存储装置的数量被检测以表现可变 I/O 响应时间。在块 1004 中, 确定装置组内的可同时繁忙或者表现可变响应时间的装置的最大数量。该最大数量可被称为目标数量。在一个实施例中, 存储装置是由于执行写入请求、擦除请求或高速缓存刷新可表现可变响应时间的 SSD。在一个实施例中, 选择目标数量, 使得仍可执行重构读取。

[0066] 在一个实施例中, I/O 调度器可检测保证将目标数量增加到重构读取不再有效的水平的条件。例如, 给定装置的未决写入请求的数量可达到等待阈值(即, 写入请求已对于明显的时间未决, 并且确定它们应不再等待)。作为替代方案, 可检测可具有相对高优先级的给定数量的写入请求, 如上所述, 该相对高优先级为了后面的发出不能被累积。如果 I/O 调度器检测到这种条件(条件块 1006), 那么, 在块 1008 中, I/O 调度器可基于一个或多个检测条件增或减目标。例如, 如果适量的高优先级写入请求是未决的, 或者出现一些其它的条件, 那么, I/O 调度器可允许目标超过支持的冗余量。在块 1010 中, I/O 调度器可确定装置组内的 N 个存储装置表现可变 I/O 响应时间。如果 N 大于目标(条件块 1012), 那么在块 1014 中, 存储装置可以以减小 N 的方式减小。否则, 在块 1016 中, I/O 调度器可以以提高性能的方式调度请求。例如, 如后面描述的那样, I/O 调度器可利用重构读取请求的能力。

[0067] 现在参照图 11, 该图表示用于减少表现可变 I/O 响应时间的存储装置的数量方法 1100 的一个实施例。本实施例中的步骤以依次的方式被表示。但是, 一些步骤可以以与所示的次序不同的次序出现, 一些步骤可同时被执行, 一些步骤可与其它步骤组合, 并且, 在另一实施例中可缺少一些步骤。

[0068] 在块 1102 中, I/O 调度器可确定减少执行在意外时间使得可变响应时间的高等待时间操作的存储子系统内的存储装置的数量 N。在块 1104 中, I/O 调度器可选择执行高等待时间操作的给定装置。在块 1106 中, I/O 调度器可停止在给定装置上执行高等待时间操作, 并将 N 减 1。例如, I/O 调度器可停止向给定的存储装置发出写入请求和擦除请求。另外, 相应的 I/O 调度器可停止执行发出的写入请求和擦除请求。在块 1108 中, I/O 调度器可开始在给定的装置上执行低等待时间操作, 诸如读取请求。这些读取请求可包含重构读取请求。以这种方式, 装置留下长的等待时间响应状态, 并且 N 减小。

[0069] 现在参照图 12, 该图表示用于在共享的数据存储上用有效的等待时间保持读取操作的一个实施例。在上述的网络构架 100 和数据存储阵列 120a ~ 120b 中体现的部件可一般根据该方法操作。本实施例中的步骤以依次的方式被表示。但是, 一些步骤可以以与所

示的次序不同的次序出现,一些步骤可同时被执行,一些步骤可与其它步骤组合,并且,在另一实施例中可缺少一些步骤。

[0070] 图 12 的方法可代表执行方法 1000 中的步骤 1016 所采取的步骤的一个实施例。在块 1201 中, I/O 调度器接收引向表现可变响应时间行为的第一装置的原读取请求。第一装置可能由于接收特定的调度操作(即,已知的原因)或者由于一些未知的原因正表现可变响应时间。在各种实施例中,可至少部分地基于给定操作的期望的等待时间确定考虑的可变响应时间。例如,基于装置的特性和 / 或操作的最近的历史,可望在给定的时间周期内出现对于给定装置的响应。例如,可用被确定为反映可接受响应等待时间的范围的 Δ 对于装置确定平均响应等待时间。这种 Δ 可被选择为应对 99% 的交易或任何其它适当数量的交易。如果在期望的时间周期内不接收响应,那么可触发重构读取的启动。

[0071] 一般而言,是否模仿重构读取基于成本益处分析,该成本益处分析比较与执行重构读取相关的成本与获得重构读取的结果的(潜在)益处。例如,如果在给定的时间周期内不接收对于给定装置内的原读取请求的响应,那么可以预测,装置正在执行将使得超过要启动的重构读取的操作。因此,重构读取可被启动。可采取这种动作以(例如)保持读取服务性能的给定水平。注意,当确定是否启动启动重构读取时,也可考虑其它的因素,诸如当前的负载、接收的请求的类型、请求的优先级、系统中的其它装置的状态和图 7 和图 8 描述的各种特性等。并且,注意,虽然可由于用于原读取的相对长的响应等待时间启动重构读取,但是,期望原读取请求将事实上完成。事实上,原读取和重构读取可成功地完成并提供结果。因此,为了服务原请求,不需要重构读取。这与由于诸如检测等待时间和表示交易将(或者可能)不成功完成的错误的一些指示的错误条件使得的等待时间相反。例如,由于不能读取给定的存储位置使得的装置超时代表不期望完成的响应。在这些情况下,为了服务请求,可能需要重构读取。因此,在各种实施例中,系统可有效地包括用于给定的装置的至少两个超时条件。第一超时与在其之后即使未必需要也可启动重构读取的时间周期对应。以这种方式,重构读取可作为非错误相关调度处理的正常部分被加入调度算法中。在第一超时之后出现的第二超时代表在其之后认为出现了错误条件的的时间周期。在这种情况下,由于期望不通过指示错误的装置服务原读取,因此,也可启动重构读取。

[0072] 鉴于以上情况, I/O 调度器可然后确定是否要启动与原读取对应的重构读取(决定块 1202)。重构读取一般需要由第一装置以外的装置服务的一个或多个读取。在确定是否要启动重构读取时,可考虑许多因素。一般而言, I/O 调度器进行成本 / 益处分析以确定尝试用第一装置服务原读取或者尝试通过发出重构读取服务原读取是否会“更好”。如上所述,当确定是否启动重构读取时,可以考虑大量的因素。在给定的情况下“更好”的情况可改变,可被编程,并且可被动态确定。例如,算法可以是,使得它总是有利于更快的读取响应时间。在这种情况下,可确定是否可(或者可能)在通过原装置服务原读取之前完成重构的服务。作为替代方案,算法可确定在给定的时间有利于降低系统负载。在这种情况下, I/O 调度器可选择不用其附加的经费启动重构读取—即使重构读取可比原读取更快地完成。并且,在这种确定中,可以使用速度与经费的更细微的平衡。在各种实施例中,算法可通过初始加权(例如,不管速度如何,总是偏好速度)被编程。这种加权可以以恒定的,可被编辑以根据各种条件动态改变。例如,条件可包含一天的时间、接收的 I/O 请求的比率、接收的请求的优先级、是否检测到特定的任务(例如,当前正在执行备份操作)和失效的检测等。

[0073] 如果调度器决定不启动重构读取,那么可通过原目标装置服务读取(块 1203)。作为替代方案,可以启动重构读取(块 1204)。在一个实施例中,选择的用于服务重构读取的其它装置是识别为表现为不可变行为的那些。通过选择表现不可变行为(例如,更容易预测行为)的装置,I/O 调度器能够更好地预测它会花费多少时间以服务重构读取。除了装置的给定的可变/不可变行为以外,I/O 调度器还可考虑各装置的其它方面。例如,在选择用于服务重构读取的特定的装置时,I/O 调度器也可评价给定装置的突出请求的量(例如,装置队列多么满)、对给定装置当前未决的请求的优先级和装置自身的期望处理速度(例如,一些装置与其它装置相比可代表较旧或者另外固有较慢的技术)等。并且,调度器可希望以在大致相同的时间返回来自装置中的每一个的相应的结果的方式调度重构读取。在这种情况下,如果预测其处理时间会与其它装置明显不同,调度器会不利于用于服务重构读取的特定的装置—即使它比其它装置快得多。大量的需要考虑的这种因素和条件是可能的,并且被设想。

[0074] 在一个实施例中,重构读取请求可继承原读取请求的优先级水平。在其它的实施例中,重构读取请求可具有与原读取请求不同的优先级。如果 I/O 调度器检测到接收相应的重构请求的选择的第二(其它)装置现在正在表现可变响应时间行为(条件块 1205)并且该第二装置被预测为保持可变直到第一装置被预测为变得不可变之后(条件块 1206),那么,在块 1208 中,I/O 调度器可向第一装置发出原读取请求。在一个实施例中,可以使用定时器以预测表现可变响应时间的存储装置什么时候可重新提供不可变响应时间。方法 1200 的控制流程通过块 C 从块 1208 移动到条件块 1212。如果第二装置不被预测为比第一装置长地保持可变(条件块 1206),那么方法 1200 的控制流程移动到块 1210。在块 1210 中,由发出的重构读取请求服务读取请求。

[0075] 如果 I/O 调度器检测到给定的可变装置变得不可变(条件块 1212),那么在块 1214 中,I/O 调度器向给定的装置发出原读取请求。I/O 调度器可指定给定的装置为不可变并且将 N(被检测为提供可变 I/O 响应时间的存储装置的数量)减 1。如果原读取请求在替代重构读取请求之前完成(条件块 1216),那么,在块 1218 中,I/O 调度器用原读取重构服务读取请求。在各种实施例中,调度器可去除重构的读取请求。作为替代方案,重构读取请求可完成并且它们的数据简单地被舍弃。否则,在块 1220 中,I/O 调度器用重构读取请求服务读取请求,并可去除原读取请求(或舍弃其返回数据)。

[0076] 注意,上述的实施例可包含软件。在这种实施例中,实现方法和/或机构的程序指令可被传输或者被存储于计算机可读介质上。被配置为存储程序指令的大量的类型的介质是可用的,并且包含硬盘、软盘、CD-ROM、DVD、快擦写存储器、可编程 ROM (PROM)、随机存取存储器(RAM)和各种其它形式的易失性或非易失性存储器。

[0077] 在各种实施例中,这里描述的方法和机构的一个或多个部分可形成云计算环境的一部分。在这种实施例中,可根据一个或多个各种模型作为服务在因特网上提供资源。这种模型可包含基础构架即服务(IaaS)、平台即服务(PaaS)和软件即服务(SaaS)。在 IaaS 中,计算机基础构架作为服务被传输。在这种情况下,计算设备一般被服务提供商拥有和操作。在 PaaS 模型中,开发人员使用以用于开发软件方案的软件工具和下层的设备可作为服务被提供,并被服务提供商托管。SaaS 一般包含服务提供商许可服务作为按需服务。服务提供商可托管软件,或者可对于给定的时间周期将软件部署到顾客。以上的模型的大量的

组合是可能的,并且被设想。另外,虽然以上的描述关注网络化的存储和控制器,但是,也可通过直接固定存储、托管操作系统和其它的方式在系统中应用上述的方法和机构。

[0078] 虽然以大量的细节描述了以上的实施例,但是,对于本领域技术人员来说,一旦完全理解以上的公开,大量的变更和修改将变得十分明显。以下的权利要求应被解释为包含所有这些变更和修改。

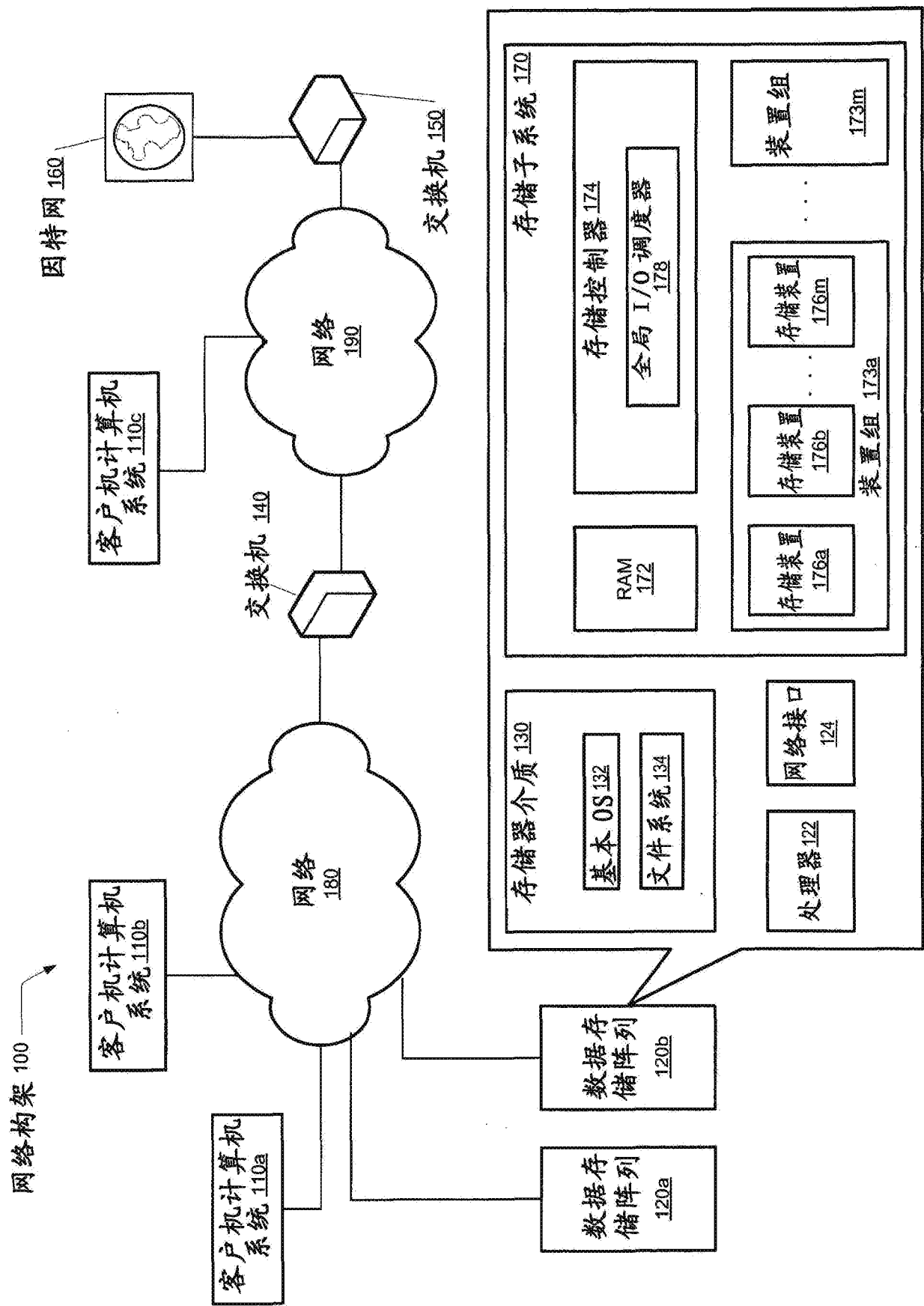


图 1

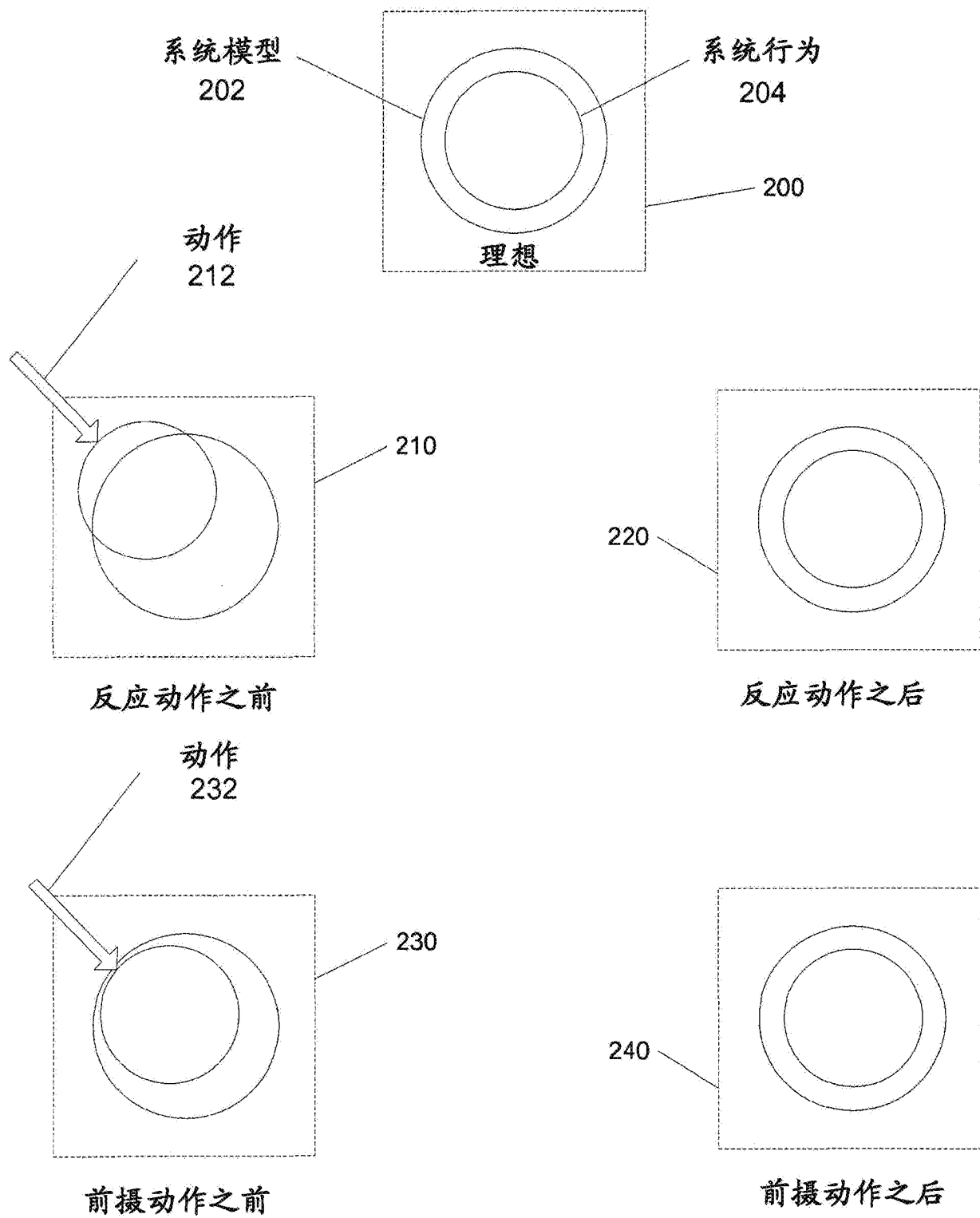


图 2

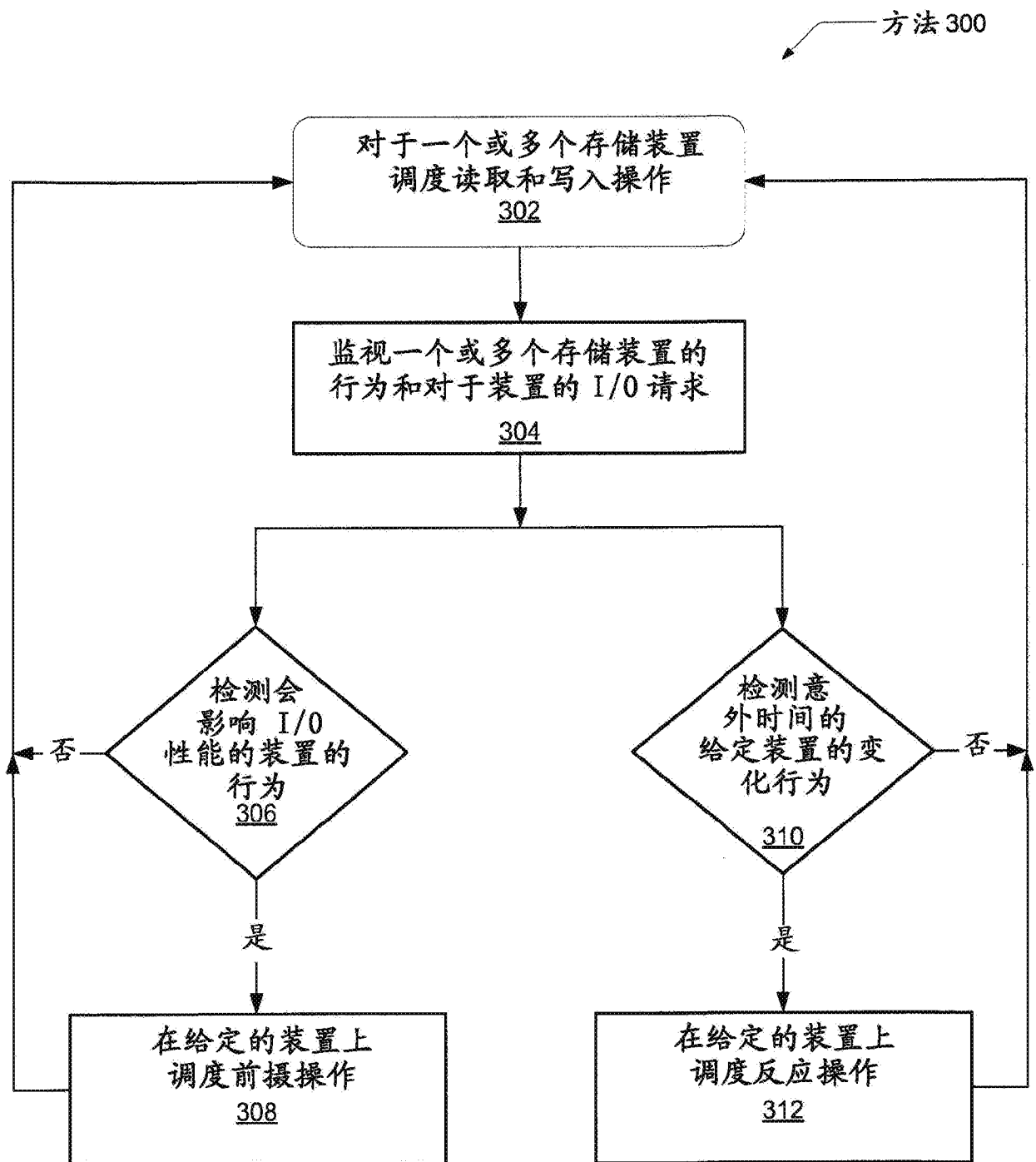


图 3

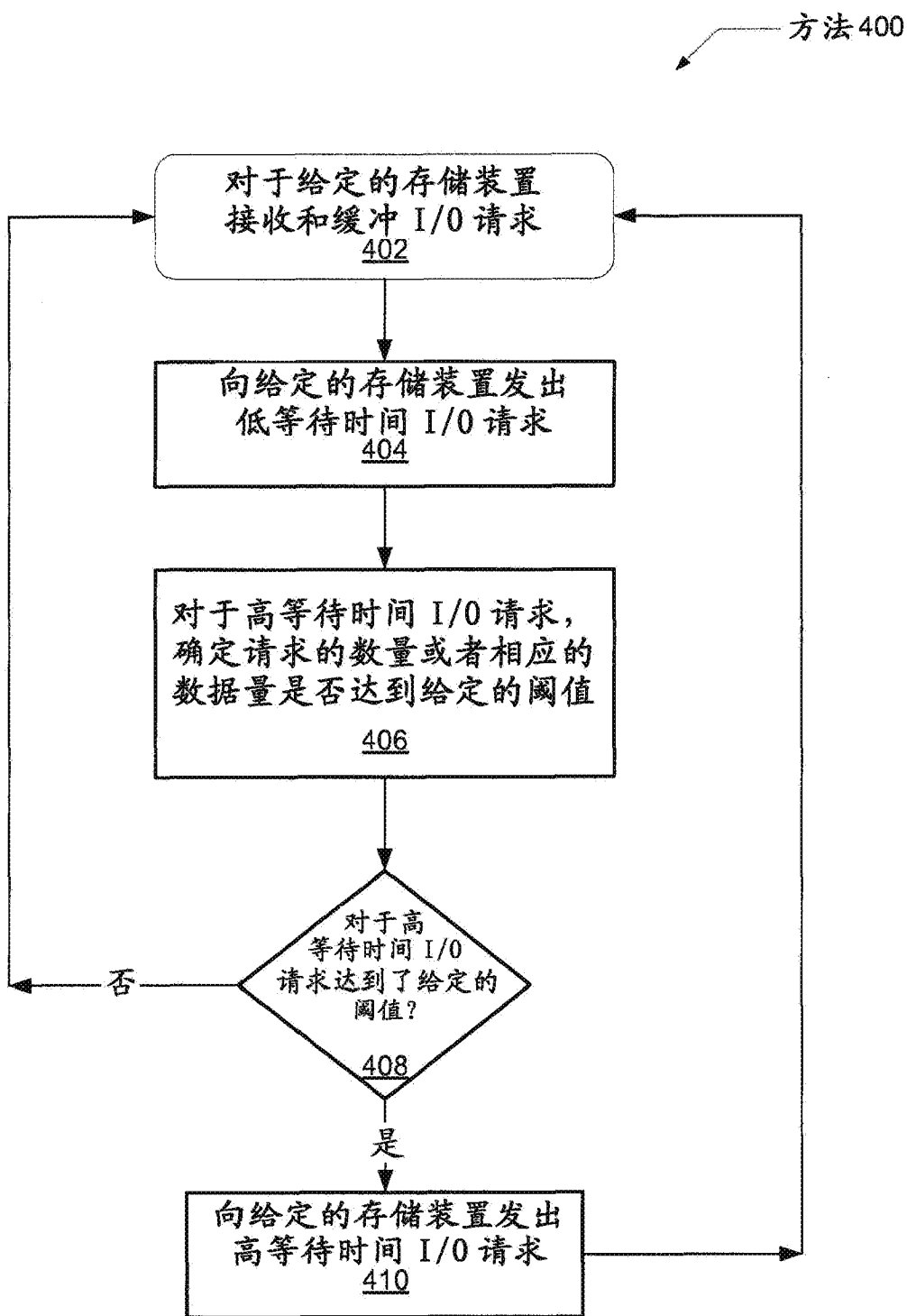


图 4

方法 500

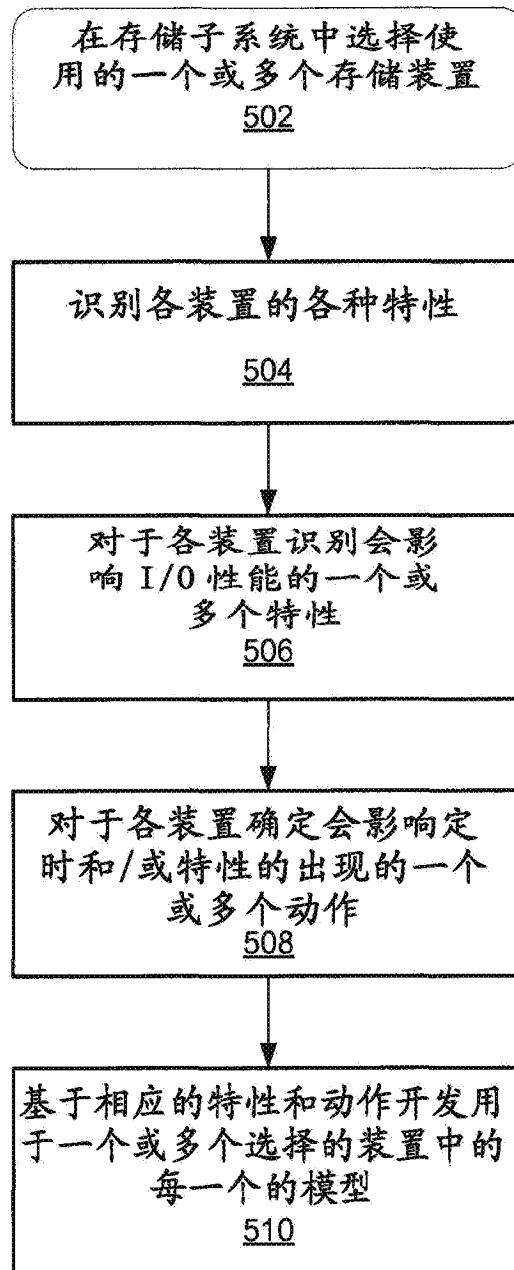


图 5

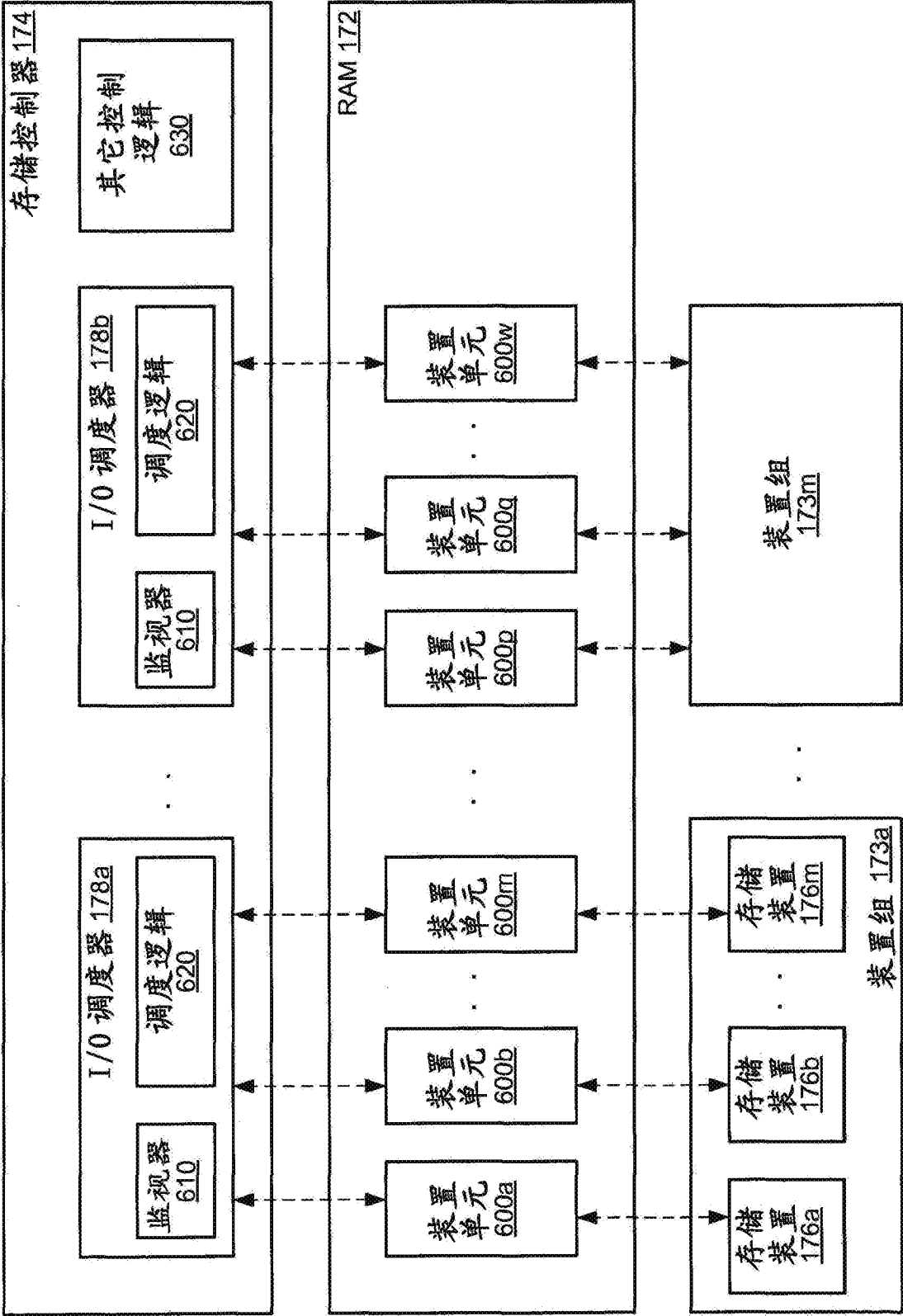


图 6

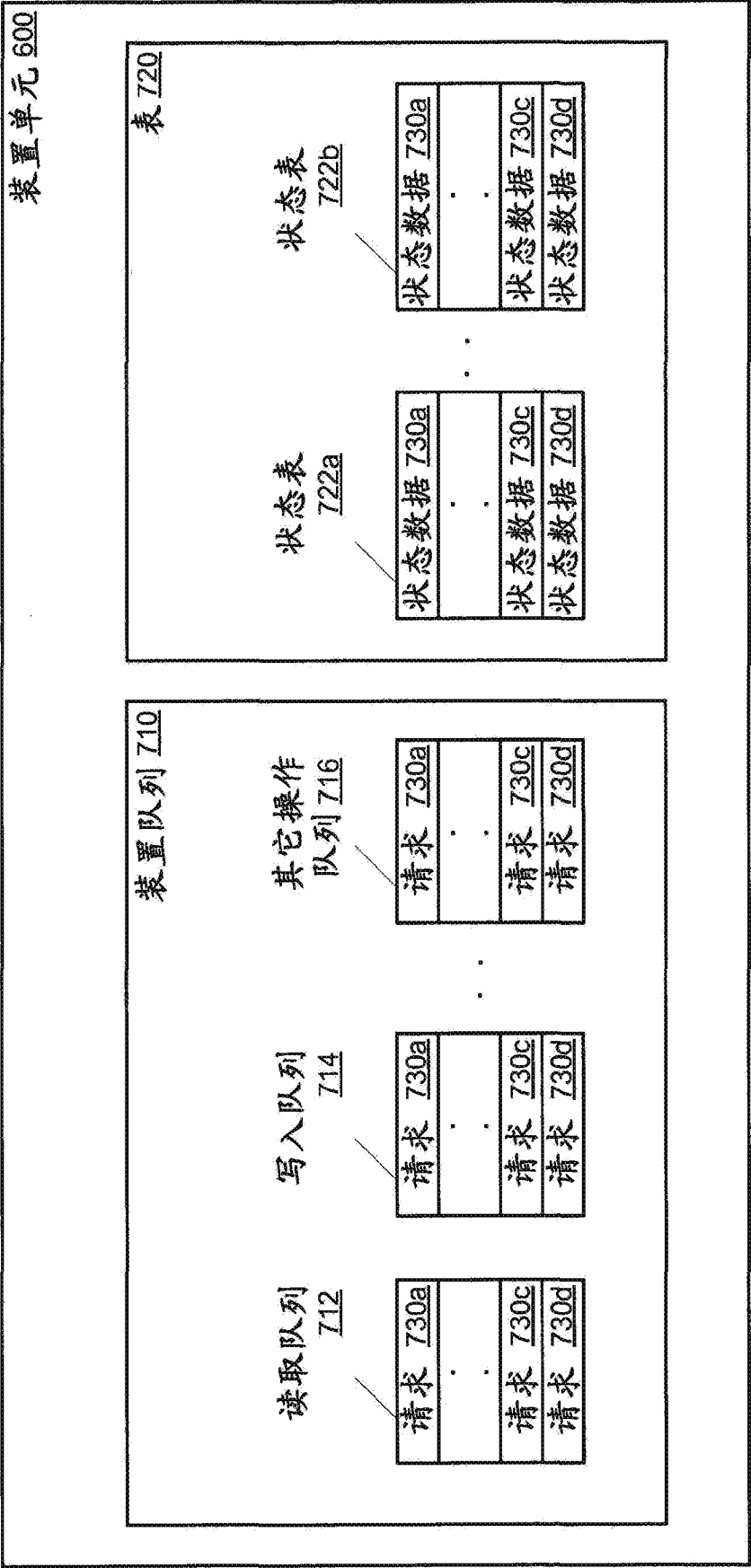


图 7

状态表 722

装置年龄 802
错误比率 804
总错误 806
可去除错误的数量 808
不可去除错误的数量 810
访问率 812
数据年龄 814
高速缓存尺寸 816
高速缓存刷新空闲时间 818
第一分配空间的分配状态 820
第二分配空间的分配状态 822
并发水平 824
期望时间 826

图 8

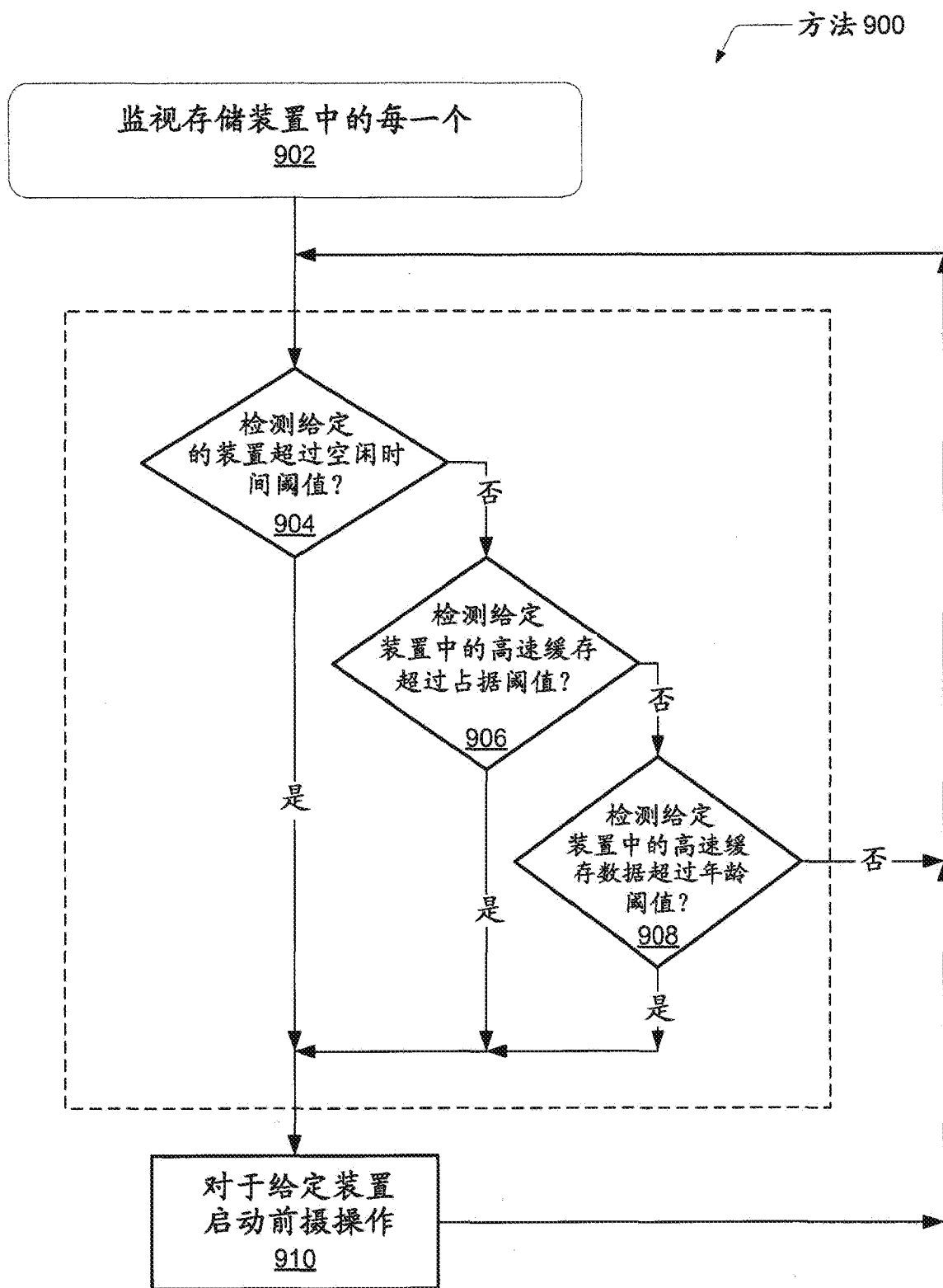


图 9

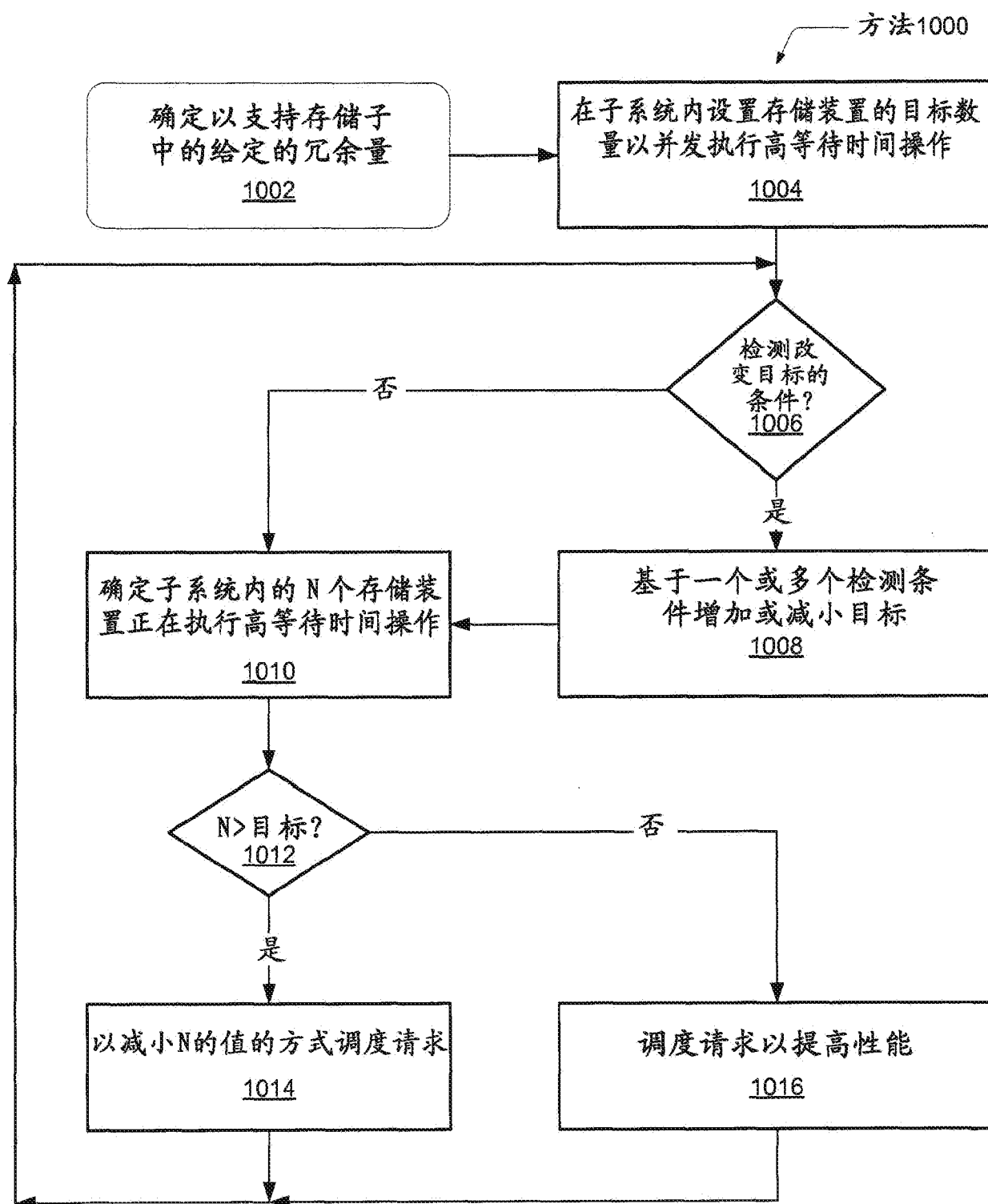


图 10

方法1100

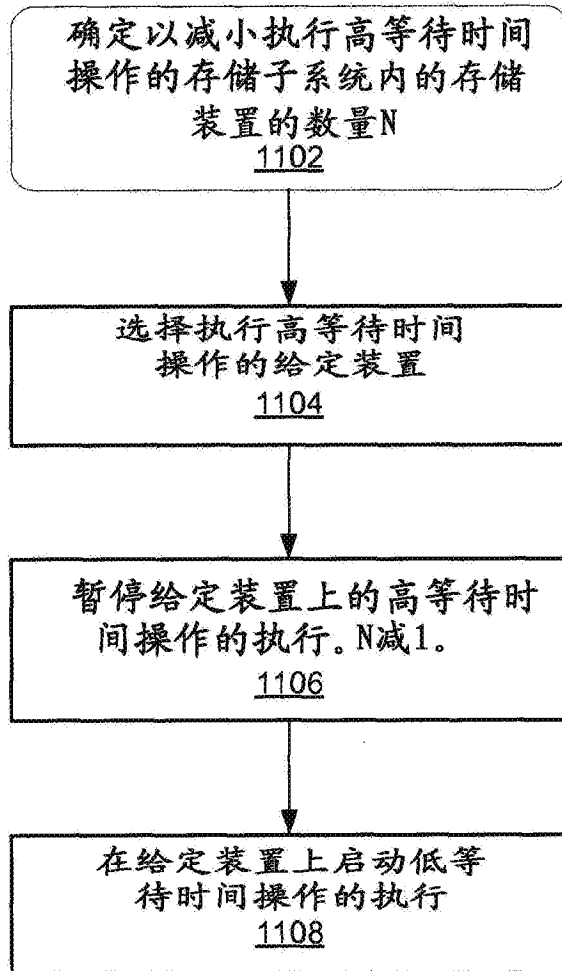


图 11

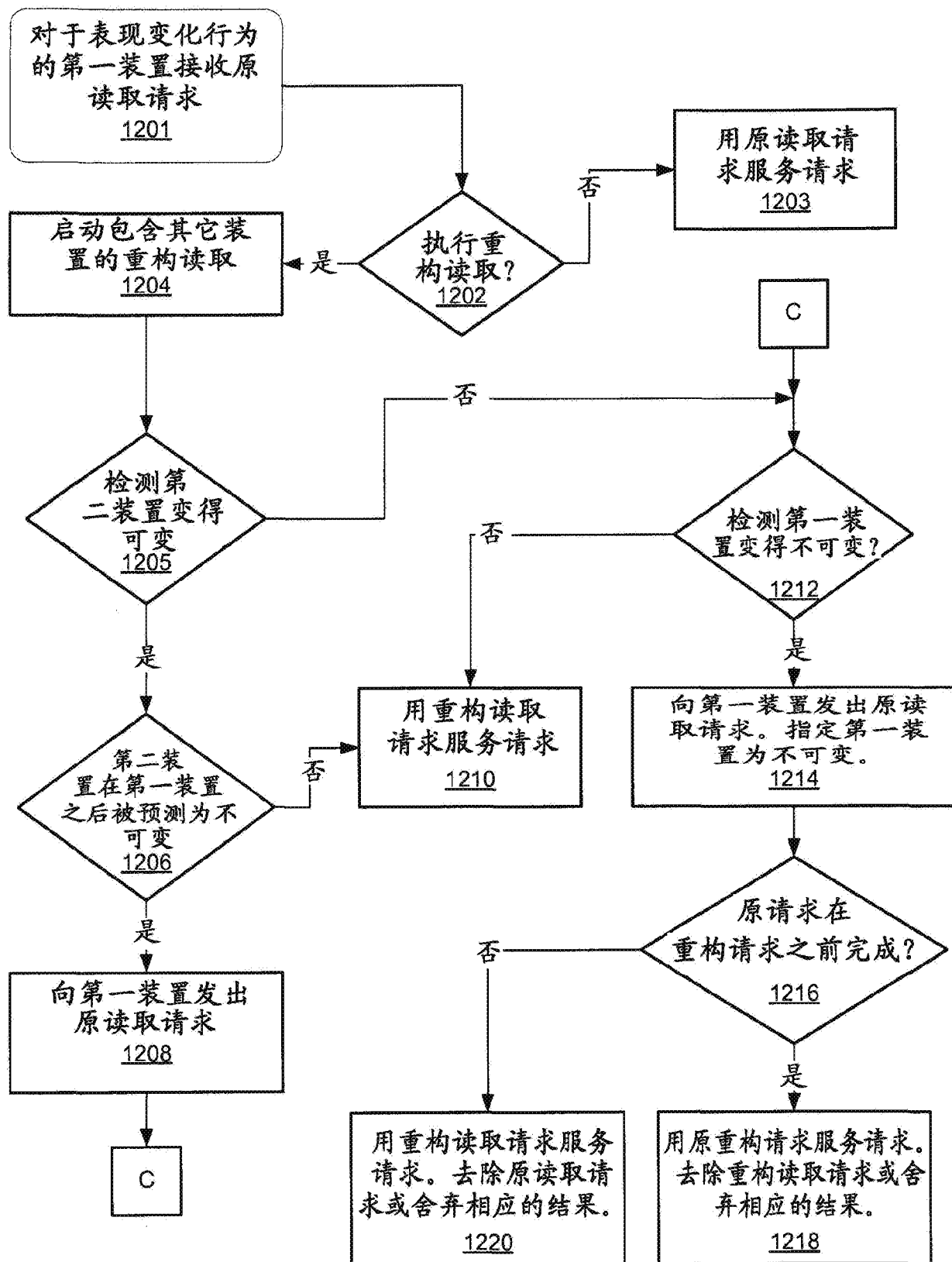


图 12