[19] 中华人民共和国国家知识产权局

[51] Int. Cl.
G06F 3/06 (2006.01)
G06F 11/14 (2006.01)

[12] 发明专利申请公布说明书

[21] 申请号 200910004738.4

[43] 公开日 2009 年 10 月 28 日

[11] 公开号 CN 101566931A

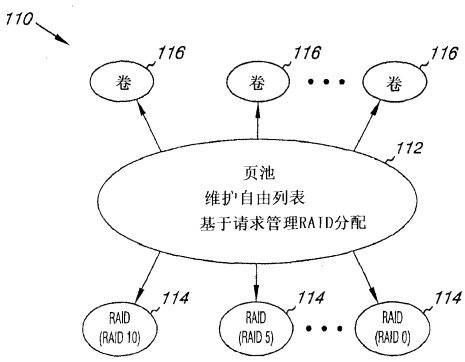
[22] 申请日 2004.8.13
[21] 申请号 200910004738.4
分案原申请号 200480026308.8
[30] 优先权
[32] 2003.8.14 [33] US [31] 60/495,204
[32] 2004.8.13 [33] US [31] 10/918,329
[71] 申请人 克姆佩棱特科技公司
地址 美国明尼苏达州
[72] 发明人 P·E·索兰 J·P·圭德
L·E·阿兹曼 M·J·克莱姆
[74] 专利代理机构 上海专利商标事务所有限公司
代理人 陈 斌
权利要求书 2 页 说明书 26 页 附图 24 页

[54] 发明名称

虚拟磁盘驱动系统和方法

[57] 摘要

提供了能够动态分配数据的磁盘驱动器系统和方法。该磁盘驱动器系统可包括含有存储池的 RAID 子系统以及含有至少一个磁盘存储系统控制器的磁盘管理器，该存储池例如维护 RAID 自由列表的存储页池或维护 RAID 空列表的磁盘存储块矩阵。RAID 子系统和磁盘管理器基于 RAID－磁盘映射跨存储池和多个磁盘驱动器动态地分配数据。RAID 子系统和磁盘管理器确定是否需要另外的磁盘驱动器，且如果需要另外的磁盘驱动器则发送通知。动态数据分配和数据分级管理允许用户在稍后需要时获取磁盘驱动器。动态数据分配也允许对用于存储的虚拟卷池的快照/时间点副本的有效数据存储，用于数据备份、恢复等的即时数据重放和数据即时融合，远程数据存储以及数据分级管理等。



1．一种磁盘驱动器系统中数据分级管理的方法，包括以下步骤：

确定多个 RAID 设备中的每一个的操作成本；

基于所述成本对所述多个 RAID 设备中的每一个进行分类；

连续检查所述多个 RAID 设备上的数据，以确定是否有数据要从一个分类的 RAID 设备移动到另一个；以及

将一个分类的 RAID 设备上存储的数据移动到另一个分类的 RAID 设备。

2．如权利要求 1 所述的方法，其特征在于，所述多个 RAID 设备中的每一个的成本基于所述 RAID 设备的效率。

3．如权利要求 2 所述的方法，其特征在于，所述多个 RAID 设备中的每一个的成本基于所述 RAID 设备的存储效率。

4．如权利要求 2 所述的方法，其特征在于，所述多个 RAID 设备中的每一个的成本基于所述 RAID 设备的物理成本。

5．如权利要求 1 所述的方法，其特征在于，所述多个 RAID 设备彼此相关地分类。

6．如权利要求 5 所述的方法，其特征在于，还包括当存储磁盘增加时，重新平衡 RAID 设备的分类。

7．如权利要求 1 所述的方法，其特征在于，连续检查所述多个 RAID 设备上的数据包括：确定所述数据的访问模式和存储成本。

8．如权利要求 1 所述的方法，其特征在于，如果数据最近未被访问，则将数据移动到较低操作成本的 RAID 设备。

9．如权利要求 1 所述的方法，其特征在于，如果数据包括历史快照数据，则将数据移动到较低操作成本的 RAID 设备。

10．如权利要求 8 所述的方法，其特征在于，如果较高操作成本的 RAID 设备上相当一部分存储空间耗尽，则将数据移动到较低操作成本的 RAID 设备。

11．如权利要求 10 所述的方法，其特征在于，如果较高操作成本的 RAID 设备上的存储空间充分耗尽，则将数据移动到较低操作成本的 RAID 设备。

12．如权利要求 1 所述的方法，其特征在于，当较低操作成本的 RAID 设备中的数据开始被更频繁地使用时，将数据移动到较高操作成本的 RAID 设备。

13．如权利要求 1 所述的方法，其特征在于，所述磁盘驱动器系统包括诸如 RAID-0、RAID-1、RAID-5 和 RAID-10 之类的多种 RAID 类型中的至少一种的存储空间。

14. 如权利要求 13 所述的方法，其特征在于，将 RAID-10 设备转换成 RAID-5 设备，以更有效地使用该 RAID 设备的物理磁盘空间。

15．如权利要求 8 所述的方法，其特征在于，当存储资源低时，积极地移动数据。

16．如权利要求 1 所述的方法，其特征在于，还包括管理存储页池，所述存储页池对每个类别的 RAID 设备包括单独的自由存储空间列表。

17．一种磁盘驱动器系统，包括：

包括存储池的 RAID 子系统；以及

具有至少一个磁盘存储系统控制器的磁盘管理器，所述磁盘存储系统控制器配置成：

确定所述 RAID 子系统的多个 RAID 设备中的每一个的操作成本；

检查所述多个 RAID 设备上的数据，以确定是否有数据要从一个 RAID 设备移动到不同操作成本的另一个 RAID 设备；以及

将一个操作成本的 RAID 设备上存储的数据移动到另一个操作成本的 RAID 设备。

18．如权利要求 17 所述的系统，其特征在于，所述 RAID 子系统还包括诸如 RAID-0、RAID-1、RAID-5 和 RAID-10 等多种 RAID 类型中的至少一种的组合。

19．如权利要求 18 所述的系统，其特征在于，还包括包含 RAID-3、RAID-4、RAID-6 和 RAID-7 的 RAID 类型。

20．一种能够进行数据分级管理的磁盘驱动器系统，包括：

确定多个 RAID 设备中的每一个的操作成本的计算装置；

基于所述成本对所述多个 RAID 设备中的每一个进行分类的分类装置；

连续检查所述多个 RAID 设备上的数据的状态检查装置，以确定是否有数据要从一个分类的 RAID 设备移动到另一个；以及

将一个分类的 RAID 设备上存储的数据移动到另一个分类的 RAID 设备的转移装置。

虚拟磁盘驱动系统和方法

本申请是申请日为2004年8月13日、国际申请号为PCT/ /US2004/026499、中国国家申请号为200480026308.8、发明名称为“虚拟磁盘驱动系统和方法”的专利申请的分案申请。

技术领域

本发明一般涉及磁盘驱动器系统和方法，尤其设计具有诸如动态数据分配和磁盘驱动器虚拟化等能力的磁盘驱动器系统。

背景技术

现有的磁盘驱动器系统是以这样的一种方式来设计的：使得虚拟卷数据存储空间与具有特定大小和位置的物理磁盘静态地相关联以供存储数据。这些磁盘驱动器系统需要了解和监视/控制数据存储空间的虚拟卷的精确位置和大小以便存储数据。另外，系统经常需要更大的数据存储空间，以便添加更多的RAID设备。然而，通常这些附加的RAID设备是昂贵的，且在实际需要额外的数据存储空间之前并不是所需的。

图14A示出了包含与具有特定大小和位置的物理磁盘相关联的虚拟卷数据存储空间以供存储、读/写和/或恢复数据的现有磁盘驱动器系统。磁盘驱动器系统基于数据存储空间的虚拟卷的特定位置和大小来静态地分配数据。结果是，将不使用清空的数据存储空间，而预先获取额外的、有时是昂贵的数据存储设备，例如RAID设备以供存储、读/写和/或恢复系统中的数据。稍后，才需要和/或使用这些额外的数据存储空间。

从而，存在对改进的磁盘驱动系统和方法的需求。还存在对有效的、动态数据分配和磁盘驱动器空间和时间管理系统和方法的需求。

发明内容

本发明提供能够动态分配数据的改进的磁盘驱动器系统和方法。该磁盘驱动

器系统可包括含有磁盘存储块的矩阵的 RAID 子系统以及含有至少一个磁盘存储系统控制器的磁盘管理器。RAID 子系统和磁盘管理器基于 RAID－磁盘映射跨磁盘存储块的矩阵和多个磁盘驱动器来动态地分配数据。RAID 子系统和磁盘管理器确定是否需要另外的磁盘驱动器，且如果需要另外的磁盘驱动器则发送通知。动态数据分配允许用户在稍后当需要时获取磁盘驱动器。动态数据分配也允许对磁盘存储块的虚拟卷矩阵或池的快照/时间点副本的有效数据存储，用于数据备份、恢复等的即时数据重放和数据即时融合，远程数据存储以及数据分级管理（data progression）等。由于将在稍后购买更便宜的磁盘驱动器，因此数据分级管理也允许推迟购买更便宜的磁盘驱动器。

在一个实施例中，提供虚拟卷或磁盘存储块的矩阵或池与物理磁盘相关联。虚拟卷或磁盘存储块的矩阵或池是由多个磁盘存储系统控制器动态地监视/控制的。在一个实施例中，每一虚拟卷的大小可以是默认的或可由用户预定义，而每一虚拟卷的位置默认为空。在分配数据之前，虚拟卷为空。可在矩阵或池的任何网格中分配数据（例如，一旦在网格中分配数据，即为该网格中的一个“点”）。一旦删除该数据，该虚拟卷再次可用，指示为“空”。因此，可在需求的基础上在稍后获取额外的数据存储空间以及有时是昂贵的磁盘存储设备，例如 RAID 设备。

在一个实施例中，磁盘管理器可管理多个磁盘存储系统控制器，多个冗余磁盘存储系统控制器可被实现来覆盖被操作的磁盘存储系统控制器上的故障。

在一个实施例中，RAID 子系统包括各 RAID 类型中至少一个的组合，RAID 类型诸如 RAID-0、RAID-1、RAID-5 和 RAID-10。可以理解，可在替换的 RAID 子系统中使用其它 RAID 类型，诸如 RAID-3、RAID-4、RAID-6 和 RAID-7 等。

本发明也提供动态数据分配方法，它包括以下步骤：提供逻辑块或磁盘存储块的默认大小，使得 RAID 子系统的磁盘空间形成磁盘存储块的矩阵；在该磁盘存储块的矩阵中写数据和分配数据；基于 RAID 子系统的磁盘空间的历史占用率确定 RAID 子系统的磁盘空间的占用率；确定是否需要额外的磁盘驱动器；且如果需要额外的磁盘驱动器则向 RAID 子系统发送通知。在一个实施例中，通知是通过电子邮件发送的。

本发明的磁盘驱动器系统的优点之一是，RAID 子系统能够跨虚拟数量的磁盘使用 RAID 技术。其余的存储空间可供自由使用。通过监视存储空间和确定 RAID 子系统的存储空间的占用率，用户不必获取昂贵但购买时无用的大量驱动器。因此，当实际需要驱动器时添加驱动器以满足存储空间的渐增需求将显著地减少磁盘驱

动器的总成本。同时，基本上改进了对磁盘使用的效率。

本发明的另一优点是，该磁盘存储系统控制器对任何计算机文件系统是通用的，而不仅用于特定计算机文件系统。

本发明也提供数据即时重放的方法。在一个实施例中，数据即时重放方法包括以下步骤：提供逻辑块或磁盘存储块的默认大小，使得 RAID 子系统的磁盘空间形成存储页池或磁盘存储块的矩阵；以预定的时间间隔自动生成存储页池的卷的快照或磁盘存储块的矩阵的快照；以及存储存储页池或磁盘存储块的矩阵的快照或增量的地址索引，使得磁盘存储块矩阵的快照或增量可通过所存储的地址索引来即时定位。

数据即时重放方法以用户定义的时间间隔、用户配置的动态时戳（例如，每隔几分钟或几小时等）或由服务器指示的时间自动生成 RAID 子系统的快照。万一出现系统故障或病毒攻击，这些加时戳的虚拟快照允许大约数分钟或小时内等的数据即时重放和数据即时恢复。该技术也被称为即时重放融合，即，及时地融合崩溃或攻击前不久的数据，且可即时使用崩溃或攻击之前所存储的快照用于将来的操作。

在一个实施例中，可在本地 RAID 子系统或在远程 RAID 子系统中存储快照，使得如果由于例如恐怖袭击等而发生主要系统崩溃时，数据的完整性不受影响，且可即时恢复数据。

数据即时重放方法的另一优点是，快照可用于测试，而同时系统保持其操作。实时数据可用于实时测试。

本发明也提供数据即时重放的系统，它包括 RAID 子系统和具有至少一个磁盘存储系统控制器的磁盘管理器。在一个实施例中，RAID 子系统和磁盘管理器基于 RAID—磁盘映射跨多个驱动器的磁盘空间自动分配数据，其中 RAID 子系统的磁盘空间形成磁盘存储块的矩阵。磁盘存储系统控制器以预定的时间间隔自动生成磁盘存储块的矩阵的快照，并存储磁盘存储块的矩阵的快照或增量的地址索引，使得可通过所存储的地址索引即时定位磁盘存储块的矩阵的该快照或增量。

在一个实施例中，磁盘存储系统控制器从磁盘存储块的矩阵的快照中监视数据使用的频率，并应用老化规则，使得较少使用或访问的数据被移至较不昂贵的 RAID 子系统中。类似地，当位于较不昂贵的 RAID 子系统中的数据要被更频繁地使用时，控制器将该数据移动至较昂贵的 RAID 子系统中。从而，用户能够选择所期望的 RAID 子系统公文包来满足其自身的存储需求。从而，磁盘驱动器系统的成

本可显著地减少，并由用户动态控制。

通过以下详细描述，对本领域的技术人员而言，本发明的这些和其它特征和优点将是显而易见的，在详细描述中示出和描述了本发明的说明性实施例，包括用于实施本发明的所构想的最佳模式。可以认识到，本发明可在各种明显的方面中修改，但均不背离本发明的精神和范围。从而，附图和详细描述将被示为本质上是说明性而非限制性的。

附图说明

图 1 示出了根据本发明的原理的计算机环境中的磁盘驱动器系统的一个实施例。

图 2 示出了根据本发明的原理，具有用于磁盘驱动器的 RAID 子系统的存储页池的动态数据分配的一个实施例。

图 2A 示出了磁盘驱动器系统的 RAID 子系统中的常规数据分配。

图 2B 示出了根据本发明的原理的磁盘驱动器系统的 RAID 子系统中的数据分配。

图 2C 示出了根据本发明的原理的动态数据分配方法。

图 3A 和 3B 是根据本发明的原理，RAID 子系统的磁盘存储块在多个时间间隔处的快照的示意图。

图 4 是根据本发明的原理，通过使用 RAID 子系统的磁盘存储块的快照的数据即时融合功能的示意图。

图 5 是根据本发明的原理，通过使用 RAID 子系统的磁盘存储块的快照的本地一远程数据复制和即时重放功能的示意图。

图 6 示出了根据本发明的原理，使用同一 RAID 接口来执行 I/O 和将多个 RAID 设备串接成卷的快照的示意图。

图 7 示出了根据本发明的原理的快照结构的一个实施例。

图 8 示出了根据本发明的原理的 PITC 生存周期的一个实施例。

图 9 示出了根据本发明的原理，具有多级索引的 PITC 表结构的一个实施例。

图 10 示出了根据本发明的原理的 PITC 表的恢复的一个实施例。

图 11 示出了根据本发明的原理，具有自有页序列以及非自有页序列的写进程的一个实施例。

图 12 示出了根据本发明的原理的示例性快照操作。

图 13A 示出了含有与具有特定大小和位置的物理磁盘相关联的虚拟数据存储空间以供静态分配数据的现有磁盘驱动器系统。

图 13B 示出了图 13A 的现有磁盘驱动器系统中的卷逻辑块映射。

图 14A 示出了根据本发明的原理，含有磁盘块虚拟卷矩阵以供动态分配系统中的数据的磁盘驱动器系统的一个实施例。

图 14B 示出了在如图 14A 中所示的磁盘存储块虚拟卷矩阵中的动态数据分配的一个实施例。

图 14C 示出了根据本发明的原理，存储虚拟卷页池的一个实施例的卷－RAID 页重映射的示意图。

图 15 示出了根据本发明的原理，映射到 RAID 子系统的多个磁盘存储块的三个磁盘驱动器的示例。

图 16 示出了在向如图 15 中所示的三个磁盘驱动器添加磁盘驱动器之后，磁盘驱动器存储块的重映射的示例。

图 17 示出了根据本发明的原理的数据分级管理操作中的可访问数据页的一个实施例。

图 18 示出了根据本发明的原理的数据分级管理操作的一个实施例的流程图。

图 19 示出了根据本发明的原理的压缩页布局的一个实施例。

图 20 示出了根据本发明的原理的高级磁盘驱动器系统中的数据分级管理的一个实施例。

图 21 示出了根据本发明的原理的子系统中的外部数据流的一个实施例。

图 22 示出了子系统中的内部数据流的一个实施例。

图 23 示出了独立维护相干性的每一子系统的一个实施例。

图 24 示出了根据本发明的原理的混合 RAID 瀑布式数据分级管理的一个实施例。

图 25 示出了根据本发明的原理，存储页池的多个自由列表的一个实施例。

图 26 示出了根据本发明的原理的数据库示例的一个实施例。

图 27 示出了根据本发明的原理的 MRI 映像示例的一个实施例。

具体实施方式

本发明提供能够动态分配数据的改进的磁盘驱动器系统和方法。磁盘驱动器

系统可包括含有维护RAID自由列表的存储页池或者磁盘存储块矩阵的RAID子系统，以及含有至少一个磁盘存储系统控制器的磁盘管理器。RAID子系统和磁盘管理器基于 RAID－磁盘映射跨存储页池或磁盘存储块矩阵和多个磁盘驱动器动态地分配数据。RAID子系统和磁盘管理器确定是否需要另外的磁盘驱动器，且如果需要另外的磁盘驱动器则发送通知。动态数据分配允许用户当稍后需要磁盘驱动器时获取磁盘驱动器。动态数据分配也允许对磁盘存储块的虚拟卷矩阵或池的快照/时间点副本的有效数据存储，用于数据备份、恢复等的即时数据重放和数据即时融合，远程数据存储以及数据分级管理等。由于可稍后购买较廉价磁盘驱动器，数据分级管理也允许推迟购买较廉价的磁盘驱动器。

图 1 示出了根据本发明的原理的计算机环境 102 中的磁盘驱动器系统 100 的一个实施例。如图 1 中所示，磁盘驱动器系统 100 包括 RAID 子系统 104 和具有至少一个磁盘存储系统控制器（图 16）的磁盘管理器 106。RAID 子系统 104 和磁盘管理器 106 基于 RAID－磁盘映射跨多个磁盘驱动器 108 的磁盘空间动态分配数据。另外，RAID 子系统 104 和磁盘管理器 106 能够基于跨磁盘空间的数据分配来确定是否需要另外的磁盘驱动器。如果需要另外的磁盘驱动器，则向用户发送通知，使得如果期望则可添加另外的磁盘空间。

根据本发明的原理，在一个实施例中，在图 2 中示出了具有动态数据分配（或称为“磁盘驱动器虚拟化”）的磁盘驱动器系统 100，在另一实施例中，在图 14A 和 14B 中示出了该系统。如图 2 中所示，磁盘存储系统 110 包括存储页池 112，即包含可自由存储数据的数据存储空间列表的数据存储池。页池 112 维护 RAID 设备 114 的自由列表，并基于用户请求管理读/写分配。将用户所请求的磁盘存储卷 116 发送给页池 112 以获取存储空间。每一卷可请求具有相同或不同 RAID 等级（例如，RAID 10、RAID 5、RAID 0 等）的相同不同的存储设备类。

本发明的动态数据分配的另一实施例在图 14A 和 14B 中示出，其中根据本发明的原理，含有多个磁盘存储系统控制器 1402 和由该多个磁盘存储系统控制器 1402 控制的磁盘存储块 1404 的矩阵的磁盘存储系统 1400 动态分配系统中的数据。提供虚拟卷或块 1404 的矩阵用于与物理磁盘相关联。虚拟卷或块 1404 的矩阵是由多个磁盘存储系统控制器 1402 动态监视/控制的。在一个实施例中，可预定义每一虚拟卷 1404 的大小，例如 2M 字节，每一虚拟卷 1404 的位置默认为空。在分配数据之前，虚拟卷 1404 中的每一个皆为空。可在矩阵或池的任何网格中分配数据（例如，一旦在网格中分配了数据，即为该网格中的一个“点”）。一旦删除数据，该

虚拟卷 1404 再次可用，指示为“空”。因此，可在需求的基础上在稍后获取额外的且有时是昂贵的磁盘存储设备，例如 RAID 设备。

从而，RAID 子系统能够跨虚拟数量的磁盘使用 RAID 技术。其余的存储空间可供自由使用。通过监视存储空间和确定 RAID 子系统的存储空间的占用率，用户不必获取昂贵但购买时无用的大量驱动器。因此，当实际需要驱动器时添加驱动器以满足存储空间的渐增的需求将显著地减少磁盘驱动器的总成本。同时，基本上改进了对磁盘的使用效率。

同样，本发明的磁盘驱动器系统的动态数据分配允许对存储虚拟卷页池或磁盘存储块虚拟卷矩阵的快照/时间点副本的有效数据存储、用于数据恢复和远程数据存储的即时数据重放和数据即时融合、以及数据分级管理。

将在以下详细讨论由动态数据分配系统和方法及其在磁盘驱动器系统 100 中的实现所得的以上特征和优点。

动态数据分配

图 2A 示出了磁盘驱动器系统的 RAID 子系统中的常规数据分配，其中清空的数据存储空间是被俘获的且不能被分配以供数据存储。

图 2B 示出了根据本发明的原理的磁盘驱动器系统的 RAID 子系统中的数据分配，其中将可供数据存储使用的清空的数据存储混合在一起以形成页池，例如本发明的一个实施例中的单个页池。

图 2C 示出了根据本发明的原理的动态数据分配方法 200。动态数据分配方法 200 包括定义逻辑块或磁盘存储块的默认大小使得 RAID 子系统的磁盘空间形成磁盘存储块的矩阵的步骤 202；以及在其中磁盘存储块指示为“空”的该矩阵的磁盘存储块中写数据和分配数据的步骤 204。该方法还包括基于 RAID 子系统的磁盘空间的历史占用率确定 RAID 子系统的磁盘空间的占用率的步骤 206；以及确定是否需要另外的磁盘驱动器，且如果需要则向 RAID 子系统发送通知的步骤 208。在一个实施例中，通知是通过电子邮件发送的。此外，磁盘存储块的大小可设定为默认或可由用户改变。

在一个实施例中，动态数据分配有时也称为“虚拟化”或“磁盘空间虚拟化”，它每秒有效地处理大量的读和写请求。该体系结构可以要求中断处理程序直接调用高速缓存子系统。由于动态数据分配不对请求排队，它可能不能优化请求，但它可以一次拥有大量待处理请求。

动态数据分配也可以维护数据完整性且保护数据的内容以防任何控制器故障。为此，动态数据分配将状态信息写入 RAID 设备以供可靠存储。

动态数据分配还可以维护读写请求的顺序，并按照接收请求的精确顺序完成读或写请求。动态数据分配允许最大系统可用性，且支持数据至不同地理位置的远程复制。

另外，动态数据分配提供从数据讹误中恢复的能力。通过快照，用户可查看过去的磁盘状态。

动态数据分配管理 RAID 设备并提供存储抽象以创建和扩充大型设备。

动态数据分配向服务器呈现虚拟磁盘设备；该设备被称为卷。对服务器而言，卷一样工作。它可返回对序号的不同信息，但卷基本上如同磁盘驱动器一样工作。卷提供对多个 RAID 设备的存储抽象以创建更大的动态卷设备。卷包括多个 RAID 设备，以供对磁盘空间的有效使用。

图 21 示出了现有的卷逻辑块映射。图 14C 示出了根据本发明的原理，存储虚拟卷页池的一个实施例的卷－RAID 页的重映射。每一卷被分成一组页，例如 1、2、3 等，且每一 RAID 被分成一组页。在一个实施例中，卷页大小和 RAID 页大小可以相同。从而，本发明的卷－RAID 页映射的一个示例是使用 RAID-2 的页#1 被映射到 RAID 页#1。

动态数据分配维护卷的数据完整性。数据被写入卷中，并向服务器确认。数据完整性覆盖各种控制器配置，包括独立和通过控制器故障的冗余。控制器故障包括电源故障、电源循环、软件异常和硬复位。动态数据分配一般不处理由 RAID 覆盖的磁盘驱动器故障。

动态数据分配为控制器提供最高级的数据抽象。它从前端接受请求，并最终使用 RAID 设备将数据写入磁盘。

动态数据分配包括各种内部子系统：

- 高速缓存——通过向服务器提供快速响应时间以及将写捆绑至数据插件来平滑对卷的读和写操作。
- 配置——包含创建、删除、检索和修改数据分配对象的方法。提供组件用于为较高级系统应用程序创建工具箱。
- 数据插件——取决于卷配置，将卷读和写请求分发给各个子系统。
- RAID 接口——向用户和其它动态数据分配子系统提供 RAID 设备抽象以创建更大卷。

- 复制/镜像/交换——将卷数据复制到本地和远程卷。在一个实施例中，可仅复制由服务器写的块。
- 快照——提供数据的增量式卷恢复。它即时地创建过去卷状态的视图卷（View Volume）。
- 代理卷——实现至远程目的卷的请求通信，用于支持远程复制。
- 记帐——向用户就分配的存储、活动、性能以及数据恢复索要费用。

动态数据分配也将配置中的任何错误和显著改变记录日志中。

图 21 示出了该子系统中的外部数据流的一个实施例。外部请求来自前端。请求包括，获取卷信息、读和写。所有的请求都含有卷 ID。卷信息是由卷配置子系统来处理的。读和写请求包含 LBA。写请求也包含数据。

取决于卷配置，动态数据分配将请求传递给多个外部层。远程复制将请求传递给前端，目的地为远程目的地卷。RAID 接口将请求传递给 RAID。复制/镜像/交换将请求传回给动态数据分配至目的地卷。

图 22 示出了该子系统中的内部数据流的一个实施例。内部数据流以高速缓存开始。高速缓存可将写请求置于高速缓存中或将请求直接传递给数据插件。高速缓存支持从前端 HBA 设备的直接 DMA。可快速完成请求，并将响应返回给服务器。数据插件管理器是高速缓存下方请求流的中心。对每一卷，它为每一请求调用所注册的子系统对象。

影响数据完整性的动态数据分配子系统可要求对控制器相干性的支持。如图 23 中所示，每一子系统独立维护相干性。相干性更新避免跨相干性链路复制数据块。高速缓存相干性可要求将数据复制到对等控制器。

磁盘存储系统控制器

图 14A 示出了根据本发明的原理，含有多个磁盘存储系统控制器 1402 和由多个磁盘存储系统控制器 1402 控制的磁盘存储块或虚拟卷 1404 的矩阵以供动态分配系统中的数据的磁盘存储系统 1400。图 14B 示出了在磁盘存储块或虚拟卷 1404 的虚拟卷矩阵中动态数据分配的一个实施例。

在一个操作中，磁盘存储系统 1400 以预定的时间间隔自动生成磁盘存储块或虚拟卷 1404 的矩阵的快照，并存储磁盘存储块或虚拟卷 1404 的矩阵的该快照或其中的增量的地址索引，使得磁盘存储块或虚拟卷 1404 的矩阵的快照或增量可通过所存储的地址索引来即时定位。

在另一个操作中，磁盘存储系统控制器 1402 从磁盘存储块 1404 的矩阵的快照中监视数据使用的频率，并应用老化规则，使得较少使用或访问的数据被移至较不昂贵的 RAID 子系统中。类似地，当位于较不昂贵的 RAID 子系统中的数据开始更频繁使用时，控制器将该数据移动至较昂贵的 RAID 子系统中。从而，用户能够选择所期望的 RAID 子系统公文包来满足其自身的存储需求。从而，磁盘驱动系统的成本可显著地减少，并由用户动态控制。

RAID－磁盘映射

RAID 子系统和磁盘管理器基于 RAID－磁盘映射跨多个磁盘驱动器的磁盘空间来动态分配数据。在一个实施例中，RAID 子系统和磁盘管理器确定是否需要另外的磁盘驱动器，且如果需要另外的磁盘驱动器则发送通知。

图 15 示出了根据本发明的原理，映射到 RAID-5 子系统 1500 中的多个磁盘存储块 1502-1512 的三个磁盘驱动 108（图 1）的示例。

图 16 示出了当将磁盘驱动器 1602 添加到如图 15 中所示的三个磁盘驱动器 108 之后，磁盘驱动存储块的重映射 1600 的示例。

磁盘管理器

如图 1 中所示，磁盘管理器 106 一般管理磁盘和磁盘阵列，包括分组/资源合并（pooling）、磁盘属性抽象、格式化、添加/减去磁盘、以及跟踪磁盘服务次数和出错率。磁盘管理器 106 不区分各种磁盘模型之间的差异，且为 RAID 组件提供通用的存储设备。磁盘管理器 106 也提供分组能力，该能力便于构造具有诸如 10,000 RPM 磁盘等特定特征的 RAID 分组。

在本发明的一个实施例中，磁盘管理器 106 至少是三层的：抽象、配置和 I/O 优化。磁盘管理器 106 向较高层呈现“磁盘”，较高层可以是例如，本地或远程附加的物理磁盘驱动器或远程附加的磁盘系统。

常见的基础特征是，这些设备中的任何一个可以是 I/O 操作的目标。抽象服务为较高层（尤其是 RAID 子系统）提供统一数据路径接口，且为管理员管理目标设备提供通用的机制。

本发明的磁盘管理器 106 也提供分组能力以简化管理和配置。磁盘可被命名且被置于组中，组也可被命名。分组是简化诸如将卷从磁盘的一个分组迁移至另一个、将磁盘的一分组专用于特定功能、指定磁盘的分组为备用等任务的强大特征。

磁盘管理器也与诸如负责检测外部设备存在与否的SCSI设备子系统等设备接口。SCSI 设备子系统至少对于光纤通道/SCSI 类型设备而言能够确定作为块类型目标设备的设备的子集。正是这些设备是由磁盘管理器管理和抽象的。

此外，磁盘管理器负责响应于来自 SCSI 设备层的流程控制。磁盘管理器拥有排队的能力，这提供了将 I/O 请求作为方法聚集以优化磁盘驱动器系统吞吐量的机会。

而且，本发明的磁盘管理器管理多个磁盘存储系统控制器。同样，可实现多个冗余磁盘存储系统控制器来覆盖所操作的磁盘存储系统控制器的故障。冗余磁盘存储系统控制器也是由磁盘管理器管理的。

磁盘管理器与其它子系统的关系

磁盘管理器与若干其它子系统交互。RAID 子系统是由磁盘管理器为数据路径活动提供的服务的主要客户机。RAID 子系统将磁盘管理器用作到用于 I/O 的磁盘的排他路径。RAID 系统也监听来自磁盘管理器的事件，以确定磁盘的存在和操作状态。RAID 子系统也与磁盘管理器一起工作来为 RAID 设备的构造分配范围。管理控制监听磁盘事件，以了解磁盘的存在以及了解操作状态改变。在本发明的一个实施例中，RAID 子系统 104 可包括至少一个 RAID 类型的组合，RAID 类型诸如 RAID-0、RAID-1、RAID-5 和 RAID-10。可以理解，可在替换的 RAID 子系统中使用其它 RAID 类型，诸如 RAID-3、RAID-4、RAID-6 和 RAID-7 等。

在本发明的一个实施例中，磁盘管理器利用配置访问服务来存储持久配置和诸如对表示层的统计等当前过渡性的只读信息。磁盘管理器向配置访问注册处理程序以访问这些参数。

磁盘管理器也利用 SCSI 设备层的服务来了解块设备的存在和操作状态，且含有对这些块设备的 I/O 路径。磁盘管理器向 SCSI 设备子系统查询设备，作为唯一地标识磁盘的支持方法。

数据即时重放和数据即时融合

本发明也提供数据即时重放和数据即时融合的方法。图 3A 和 3B 示出了根据本发明的原理在多个时间间隔处对 RAID 子系统的磁盘存储块的快照的示意图。图 3C 示出了数据即时重放方法 300，它包括定义逻辑块或磁盘存储块的默认大小使得 RAID 子系统的磁盘空间形成存储页池或磁盘存储块矩阵的步骤 302；以预定的

时间间隔自动生成页池的卷的快照或磁盘存储块的矩阵的快照的步骤 304；以及存储存储页池或磁盘存储块矩阵的快照或其中的增量的地址索引，使得磁盘存储块矩阵的快照或增量可通过所存储的地址索引来即时定位。

如图 3B 中所示，在每一预定时间间隔处，例如，5 分钟，诸如 T1（12:00PM）、T2（12:05PM）、T3（12:10PM）和 T4（12:15PM），自动生成存储页池或磁盘存储块矩阵的快照。存储页池或磁盘存储块矩阵的快照或其中的增量被存储在该存储页池或磁盘存储块矩阵中，使得可通过所存储的地址索引来即时定位存储页池或磁盘存储块矩阵的该快照或增量。

从而，数据即时重放方法以用户定义的时间间隔、用户配置的动态时戳（例如，每隔几分钟或几小时等）或由服务器指示的时间自动生成 RAID 子系统的快照。万一出现系统故障或病毒攻击，这些加时戳的虚拟快照允许大约数分钟或小时内等的数据的即时重放和数据的即时恢复。该技术也被称为即时重放融合，即及时地融合崩溃或攻击前不久的数据，且可即时使用崩溃或攻击之前所存储的快照用于将来的操作。

图 4 还示出了根据本发明的原理，通过使用 RAID 子系统的磁盘存储块的多个快照的数据即时融合功能 400 的示意图。在 T3 处，生成快照的平行链（parallel chain）T3'-T5'，借此由融合的数据 T3'融合和/或恢复的数据可用于替换在 T4 处将被融合的数据。类似地，可生成快照的多个平行链 T3"、T4'''，用于替换在 T4'-T5'处和 T4"-T5"处将被融合的数据。在替换实施例中，仍可将 T4、T4'-T5'、T5"处的快照存储在页池或矩阵中。

快照可存储在本地 RAID 子系统或远程 RAID 子系统，使得如果由于例如恐怖袭击而发生主要系统崩溃，数据的完整性将不受影响，且数据可被即时恢复。图 5 示出了根据本发明的原理通过使用 RAID 子系统的磁盘存储块的快照的本地一远程数据复制和即时恢复功能 500 的示意图。

远程复制执行将卷数据复制到远程系统的服务。它试图尽可能地保持本地和远程卷的紧密同步。在一个实施例中，远程卷的数据可能不能反映本地卷的数据的完美副本。网络连接和性能可能使得远程卷与本地卷不同步。

数据即时重放和数据即时融合方法的另一特征是，快照可用于测试，同时系统仍保持其操作。可使用实时数据用于实时测试。

快照和时间点副本（PITC）

根据本发明的原理，数据即时重放的一个示例是利用 RAID 子系统的磁盘存储块的快照。快照记录对卷的写操作，使得可创建视图来查看过去卷的内容。快照因此也支持通过创建对卷的先前时间点副本（PITC）的视图的数据恢复。

快照的核心实现快照的创建、聚合、管理和 I/O 操作。快照监视对卷的写，并为创建时间点副本（PITC）以通过视图卷访问。它向虚拟化层内的数据路径添加逻辑块地址（LBA）重映射层。这是 I/O 路径内的另一虚拟 LBA 映射层。PITC 可以不复制所有的卷信息，它可仅修改重映射使用的表。

快照跟踪对卷数据的改变，并提供查看来自先前时间点的卷数据的能力。快照通过为每一 PITC 维护增量写的列表来执行该功能。

快照为 PITC 简介表提供多种方法，包括：应用程序启动的和时间启动的。快照为应用程序提供创建 PITC 的能力。应用程序通过服务器上的 API 控制创建，并将创建传递给快照 API。同样，快照提供创建时间简介表的能力。

快照可以不实现日志处理系统或恢复对卷的所有写。快照可仅保存对 PITC 窗口内的单个地址的最后一次写。快照允许用户创建覆盖诸如几分钟或几小时等的所定义的短期时间的 PITC。为处理故障，快照将所有的信息写至磁盘。快照维护包含增量写的卷数据页指针。因为表提供对卷数据的映射，且如果没有它则不可访问卷数据，因此表数据必须处理控制器故障情况。

视图卷功能提供对 PITC 的访问。视图卷功能可附加于卷内除现有 PITC 以外的任何 PITC。对 PITC 的附加是相对较快的操作。视图卷功能的用途包括测试、训练、备份和恢复。视图卷功能允许写操作且不修改它所基于的基础 PITC。

在一个实施例中，设计快照以优化性能且以磁盘空间为代价而易于使用：

- 快照为用户请求提供快速响应。用户请求包括 I/O 操作、创建 PITC 和创建/删除视图卷。为此，快照使用比最小需要的更多的磁盘空间来存储表信息。对 I/O，快照将卷的当前状态概述至单张表中，使得可由单张表满足所有读和写请求。快照尽可能多地降低对正常 I/O 操作的影响。其次，对视图卷操作，快照使用与主卷数据路径相同的表机制。
- 快照最小化复制的数据量。为此，快照为每一 PITC 维护指针表。快照复制和移动指针，但它不移动卷上的数据。
- 快照使用固定大小的数据页来管理卷。跟踪个别扇区可能需要大量存储器用于单个合理大小的卷。通过使用大于扇区的数据页，某些页可包含直接从另一页复制而来的一定百分比的信息。

- 快照使用卷上的数据空间来存储数据页表。在控制器故障之后重新生成查找表。查找表分配页并进一步细分它们。
- 在一个实施例中，快照通过要求使用快照的卷在单个控制器上操作来处理控制器故障。该实施例不要求任何相干性。对卷的所有改变都记录在磁盘上或记录至可靠的高速缓存以供替换控制器恢复使用。在一个实施例中，从控制器故障中恢复要求从磁盘上读快照信息。
- 快照使用虚拟化 RAID 接口来访问存储。快照可将多个 RAID 设备作为单个数据空间使用。
- 快照支持每卷‘n’个 PITC 以及每卷‘m’个视图。对‘n’和‘m’的限制是磁盘空间和控制器存储器的函数。

卷和卷分配/布局

快照向卷添加 LBA 重映射层。重映射使用 I/O 请求 LBA 和查找表将地址转换成数据页。如图 6 中所示，使用快照的所呈现的卷与不具有快照的卷相同地运作。它具有线性的 LBA 空间并处理 I/O 请求。快照使用 RAID 接口来执行 I/O，且将多个 RAID 设备包含在卷中。在一个实施例中，快照卷的 RAID 设备的大小不是所呈现卷的大小。RAID 设备允许快照为卷内的数据页扩充空间。

一开始就启用快照的新卷仅需包括新数据页的空间。快照不创建页列表来置于底层 PITC 中。在这种情况中，底层 PITC 为空。在分配时，所有 PITC 页均位于自由列表上。通过创建--开始便启用快照的卷，它可分配比卷所呈现的更少的物理空间。快照跟踪对卷的写。在本发明的一个实施例中，将不在页池或矩阵中复制和/或存储 NULL 卷，从而提高了对存储空间的使用效率。

在一个实施例中，对这两种分配方案，PITC 均在列表的底部放置虚拟 NULL 卷。对 NULL 卷的读返回零块。NULL 卷处理之前未由服务器写的扇区。不可能发生对 NULL 卷的写。卷使用 NULL 卷用于对未写的扇区的读。

自由页的数量取决于卷的大小、PITC 的数量以及数据改变的预期速率。系统为给定的卷确定分配的页的数量。数据页的数量可随时间扩充。扩充可支持比预期更快速的数据改变、更多 PITC 或更大的卷。将新页添加至自由列表。可自动发生对自由列表添加页。

快照使用数据页来管理卷空间。每一数据页可包括几兆字节的数据。使用操作系统往往在卷的同一区域中写入多个扇区。存储器需求也指示快照使用页来管理

卷。为 1 万亿字节的卷的每一扇区维护单个 32 位指针可需要 8 吉字节的 RAM。不同的卷可具有不同的页大小。

图 7 示出了快照结构的一个实施例。快照将多个对象添加至卷结构。其它对象包括 PITC、指向活动 PITC 的指针、数据页自由列表、子视图卷以及 PITC 聚合对象。

- 活动 PITC（AP）指针由卷维护。AP 处理对卷的读和写请求的映射。AP 包含卷内所有数据的当前位置的概述。
- 数据页自由列表跟踪卷上的可用页。
- 可任选子视图卷提供对卷 PITC 的访问。视图卷包含它们自己的 AP 以记录对 PITC 的写，同时不修改基础数据。卷可支持多个子视图卷。
- 快照聚合对象为移除先前的 PITC 起见，临时链接两个 PITC。对 PITC 的聚合涉及移动数据页的所有权以及释放数据页。
- PITC 包含用于当 PITC 活动时所写的页的表和数据页。PITC 包含冻结时戳，在那刻 PITC 停止接受写请求。PITC 也包含生存时间值，该值确定何时 PITC 将聚合。

同样，在取 PITC 以提供可预测的读和写性能的时刻，快照概述整个卷的数据页指针。其它的解决方案可要求读来检查多个 PITC 以找出最新的指针。这些解决方案需要表高速缓存算法，但具有最坏情况性能。

本发明中的快照概述也减少表的最坏情况的存储器使用。它可要求将整个表加载至存储器中，但它可能仅要求加载单个表。

概述包括当前 PITC 所拥有的页，且可包括来自所有先前 PITC 的页。为确定 PITC 可写哪些页，它对每一数据页跟踪页所有权。它也对聚合进程跟踪所有权。为此，数据页指针包括页索引。

图 8 示出了 PITC 生存周期的一个实施例。每一 PITC 在作为只读提交前经过多个以下状态：

1．创建表——在创建时，表被创建。

2．提交给磁盘——这为 PITC 生成磁盘上的存储。通过在此刻写表，它保证了在取 PITC 之前分配存储表信息所需的空间。同时，也将 PITC 对象提交给磁盘。

3．接受 I/O——它成为活动 PITC（AP）——现在它为卷处理读和写请求。这是接受对表的写请求的唯一状态。PITC 生成表示它目前是活动的事件。

4．将表提交给磁盘——PITC 不再是 AP，且不再接受另外的页。新 AP 已经

接管。在此刻之后，除非在聚合操作中移除表，否则表将不再改变。它是只读的。在此刻，PITC 生成表示它被冻结且已被提交的事件。任何服务可监听该事件。

5．释放表存储器——释放表需要的存储器。该步骤也清除日志以声明所有的改变已被写入磁盘。

卷或视图卷的顶层 PITC 被称为活动 PITC（AP）。AP 满足对卷的所有读和写请求。对卷而言，AP 是可接受写请求的唯一的 PITC。AP 包含对整个卷的数据页指针的概述。

对聚合进程而言，AP 可以是目的地，而不是源。作为目的地，AP 增加所拥有的页的数量，但它不改变数据的视图。

对卷扩充，AP 立即随卷增长。新页指向 NULL 卷。非 AP PITC 对卷扩充不需要修改。

每一 PITC 维护将输入的 LBA 映射到对基础卷的数据页指针的表。该表包括指向数据页的指针。该表需要对比先前呈现的逻辑空间更多的物理磁盘空间进行寻址。图 9 示出了含有多级索引的表结构的一个实施例。该结构将卷 LBA 解码成数据页指针。如图 9 中所示，每一级对地址的越来越低的位解码。表的该结构允许快速查找并提供扩充卷的能力。对快速查找，多级索引结构使表很浅，在每一级上有多个条目。索引在每一级上执行数组查找。为支持卷扩充，多级索引结构允许添加另外的层以支持扩充。在整个情况中，卷扩充是呈现给较高层的 LBA 计数的扩充，而不是为卷分配的存储空间的实际数量的扩充。

多级索引包含整个卷数据页重映射的概述。每一 PITC 包含在提交 PITC 的时间点的卷的完整重映射列表。

多级索引结构对表的各层使用不同的条目类型。不同的条目类型支持从磁盘读信息以及在存储器中存储信息的需求。底层条目可仅包含数据页指针。顶层和中间层条目包含两个数组，一个用于下一级表条目的 LBA，另一个用于指向表的存储器指针。

当所呈现的卷大小的扩充时，先前 PITC 表的大小不需要增加，且这些表不需要修改。因为表为只读的，表中的信息可以不改变，且扩充进程通过添加指向末尾的 NULL 页指针来修改表。快照不向用户直接呈现来自先前 PITC 的表。

I/O 操作要求表将 LBA 映射到数据页指针。I/O 然后将数据页指针乘以数据页大小以获取基础 RAID 的 LBA。在一个实施例中，数据页大小是 2 的幂。

该表提供 API 来重映射 LBA、添加页以及聚合表。

快照使用数据页来存储 PITC 对象和 LBA 映射表。该表为对其表条目的 I/O 而直接访问 RAID 接口。当将该表读和写至 RAID 设备时，该表最小化修改。在没有修改的情况下，可能将表信息直接读和写入表条目结构。这减少了 I/O 所需的副本。快照可使用变更日志以阻止在磁盘上创建热点。热点是重复使用以跟踪对卷的更新的位置。变更日志记录对 PITC 表的更新以及卷的自由列表。在恢复过程中，快照使用变更日志来重新创建存储器中的 AP 和自由列表。图 10 示出了对表的恢复的一个实施例，它阐明了存储器中的 AP、磁盘上的 AP 和变更日志之间的关系。它也显示对自由列表的同样的关系。存储器中的 AP 表可从磁盘上的 AP 以及日志中重建。对任何控制器故障，通过读磁盘上的 AP 并向其应用变更日志来重建存储器中的 AP。取决于系统配置，变更日志使用不同的物理资源。对多控制器系统而言，变更日志依赖于备用电池高速缓存存储器以供存储。使用高速缓存存储器允许快照减少对磁盘写表的次数同时仍维护数据完整性。变更日志复制到备份控制器以供恢复。对单控制器系统而言，变更日志将所有信息写至磁盘。这具有在日志位置处创建磁盘上的热点的副作用。这允许将多个改变写至单个设备块。

周期性地，快照将 PITC 表和自由列表写至磁盘，从而在日志中创建检查点以及清除检查点。该周期取决于对 PITC 的更新的数量而变化。聚合进程不使用变更日志。

快照数据页 I/O 可要求请求在数据页边界之内适合。如果快照遇到跨越页边界的 I/O 请求，则它拆分该请求。它然后将请求向下传递给请求处理程序。写和读部分假定 I/O 在页边界之内适合。AP 提供 LBA 重映射以满足 I/O 请求。

AP 满足所有的写请求。快照对自有和非自有页支持两种不同的写序列。不同的写序列允许向表添加页。图 11 示出了含有自有页序列以及非自有页序列的写进程的一个实施例。

对自有页序列，该进程包括以下：

1) 找出表映射；以及

2) 自有页写——重映射 LBA，并将数据写至 RAID 接口。

之前写的页是简单的写请求。快照将数据写至该页，从而盖写当前内容。仅写 AP 所拥有的数据页。其它 PITC 所拥有的页是只读的。

对非自有页序列，该进程包括以下：

1) 找出表映射；

2) 读之前的页——执行对数据页的读，使得写请求和所读的数据构成完整的

页。这是写进程上的复制的开始。

3) 组合数据——将数据页读和写请求有效负载置于单个邻接块中。

4) 自由列表分配——从自由列表中获取新的数据页指针。

5) 将组合的数据写至新的数据页。

6) 将新页的信息提交给日志。

7) 更新表——改变表中的 LBA 重映射以反映新数据页指针。该数据页现在由该 PITC 所拥有。

添加页可要求阻塞读和写请求，直到将页添加至表中。通过将表更新写至磁盘，并为日志保存多个高速缓存的副本，快照实现控制器相干性。

就读请求而言，AP 履行所有的读请求。使用 AP 表，读请求将 LBA 重映射到数据页的 LBA。它将经重映射的 LBA 传递给 RAID 接口以满足请求。卷可履行对之前未写至卷的数据页的读请求。这些页在 PITC 表中被标记为 NULL 地址(全 1)。对该地址的请求可由 NULL 卷满足，并返回常量数据模式。由不同 PITC 所拥有的页可满足跨越页边界的读请求。

快照使用 NULL 卷来满足对之前未写的数据页的读请求。它对读取的每一扇区返回全 0。它不具有 RAID 设备或分配的空间。预期在存储器中保存全 0 的块以满足对 NULL 卷的读的数据要求。所有卷共享 NULL 卷来满足读请求。

在一个实施例中，聚合进程从卷中移除 PITC 和其自有页中的某一些。移除 PITC 创建更多的可用空间来跟踪新的差异。聚合对两个相邻的表比较差异，且仅保存较新的差异。根据用户配置，聚合周期性或手动地发生。

该进程可包括两个 PITC，源和目的地。在一个实施例中，对合格对象的规则如下：

1) 源必须是目的地之前的 PITC——源必须在目的地之前创建。

2) 目的地不可同时为源。

3) 源不可由多个 PITC 引用。当从 PITC 创建视图卷时，发生多重引用。

4) 目的地可支持多重引用。

5) AP 可以是目的地，但不可以是源。

聚合进程将所有的改变写至磁盘，且不要求任何相干性。如果控制器发生故障，卷从磁盘中恢复 PITC 信息，并重新开始聚合进程。

该进程标记两个 PITC 以供聚合，且包含以下步骤：

1) 将源状态置为聚合源——该状态被提交给磁盘以供存储器故障恢复。此

时，由于源的数据页可能无效而不再访问源。数据页可被返回至自由列表，或所有权可转移给目的地。

2) 将目的地状态置为聚合目的地——该状态被提交给磁盘以供控制器故障恢复。

3) 加载和比较表——该进程移动数据页指针。释放的数据页立即被添加至自由列表。

4) 将目的地状态置为正常——该进程完成。

5) 调整列表——将源下一指针的前一指针改为指向目的。这有效地将源从列表中移除。

6) 释放源——向自由列表返回用于控制信息的任何数据页。

以上进程支持两个 PITC 的组合。本领域的技术人员可以理解，聚合可被设计成移除多个 PITC 以及在一遍中创建多个源。

如图 2 中所示，页池维护数据页自由列表以供与该页池相关联的所有卷使用。该自由列表管理器使用来自页池的数据页将自由列表提交给永久性存储器。自由列表的更新来自多于个源：写进程分配页、控制页管理器分配页以及聚合进程返回页。

自由列表维护在某一阈值自动扩充自身的触发器。该触发器使用页池扩充方法将页添加到页池。自动扩充可以是由卷策略决定的。较重要的数据卷将被允许扩充，而较不重要的卷被强制聚合。

视图卷提供对先前时间点的访问并支持正常卷 I/O 操作。PITC 跟踪 PITC 之前的差异，视图卷允许用户访问 PITC 内包含的信息。视图卷从 PITC 中分支。视图卷支持恢复、测试、备份操作等。由于视图卷不需要数据副本，几乎即时发生视图卷的创建。视图卷可要求其自己的 AP 支持对视图卷的写。

可从当前卷 AP 中复制从卷 AP 的当前状态中取得的视图。使用 AP，视图卷允许对视图卷的写操作而无需修改基础数据。OS 可要求文件系统或文件重建来使用数据。视图卷从父卷中为 AP 和所写数据页分配空间。视图卷没有相关联的 RAID 设备信息。删除视图卷将空间释放回父卷。

图 12 示出了使用快照显示卷转移的示例性快照操作。图 12 示出了具有 10 页的卷。每一状态包含对卷的读请求履行列表。阴影块指示自有数据页指针。

从该图左侧（即，初始状态）到图中间的转移示出对页 3 和 8 的写。对页 3 的写要求改变 PITC I（AP）。PITC I 遵循新页写处理以将页 3 添加至表中。PITC 从页 J 中读取未改变的信息，并使用驱动器页 B 来存储该页。可在无需移动页的

情况下处理该 PITC 中对页 3 的所有将来的写。对页 8 的写示出了用于写至页的第二种情况。因为 PITC I 已经包含页 8，PITC I 盖写页 8 中的那部分数据。对这一情况而言，它存在于驱动器页 C 上。

从图中间到图右侧（即，最终状态）的转移示出 PITC II 与 III 的聚合。快照聚合涉及分别移除较老的页，同时仍维护两个 PITC 中的所有改变。这两个 PITC 均包含页 3 和页 8。该进程保留来自 PITC II 的较新的页并释放来自 PITC III 的页，它将页 A 和 D 返回给自由列表。

快照分配来自页池的数据页用于存储自由列表和 PITC 表信息。控制页分配对数据页进行二次分配以匹配对象所需的大小。

卷包含对控制页信息顶端的页指针。从该页中，可读取所有其它信息。

快照跟踪在某一时间间隔的使用中的页的数量。这允许快照预测用户何时需要向系统添加更多的物理磁盘空间以防止快照耗尽。

数据分级管理

在本发明的一个实施例中，数据分级管理（DP）用于将数据逐渐地移至具有适当成本的存储空间中。本发明允许用户在实际需要驱动器时添加驱动器。这将显著地减少磁盘驱动器的总成本。

数据分级管理将非最近访问的数据以及历史快照数据移至较不昂贵的存储中。对非最近访问的数据而言，这为非最近访问的任何页逐步减少了存储的成本。它可以不将数据立即移动至最低成本的存储。对历史快照数据，它将只读页移动至更有效的存储空间，诸如 RAID 5，如果该页不再由卷访问，那么将该页移动至最不昂贵的存储中。

本发明的数据分级管理的其它优点包括，维护对当前访问数据的快速 I/O 访问以及减少购买快速但昂贵的磁盘驱动器的需求。

在操作中，数据分级管理使用物理介质的成本以及用于数据保护的 RAID 设备的效率来确定存储的成本。数据分级管理也确定存储效率并相应地移动数据。例如，数据分级管理可将 RAID 10 转换成 RAID 5 设备以便更有效地使用物理磁盘空间。

数据分级管理将可访问数据定义为当前可由服务器读或写的数据。它使用可访问性来确定页应使用的存储类。如果页属于历史 PITC，则它是只读的。如果服务器在最近的 PITC 中没有更新该页，则该页仍然可访问。

图 17 示出了数据分级管理操作中的可访问数据页的一个实施例。该可访问数据页被分成以下类别：

- 最近访问的可访问——这些是卷最多使用的活动页。
- 非最近访问的可访问——最近未使用的读写页。
- 历史可访问——可由卷读的只读页——应用于快照卷
- 历史非可访问——卷当前未访问的只读数据页——应用于快照卷。快照为恢复的目的而维护这些页，且这些页一般尽可能置于最低成本的存储上。

在图 17 中，示出了快照卷的具有不同自有页的三个 PITC。由 PITC C 单独表示动态容量卷。所有这些页是可访问且读写的。这些页可具有不同的访问时间。

下表按照递增的效率或递减的金钱费用示出了各种存储设备。该存储设备的列表也按照渐慢的写 I/O 访问的大致顺序。数据分级管理计算由 RAID 设备的总物理空间划分的逻辑受保护空间的效率。

表 1：RAID 类型

类型	子类型	存储效率	写 1 块的 I/O 计数	用法
RAID 10		50%	2	具有相对较好写性能的主要读写可访问存储
RAID 5	3-驱动器	66.6%	4（2 读—2 写）	对 RAID 10 的最小效率增益，同时导致 RAID 5 写性能损失
RAID 5	5-驱动器	80%	4（2 读—2 写）	只读历史信息的优秀候选。非最近访问的可写页的良好候选。
RAID 5	9-驱动器	88.8%	4（2 读—2 写）	只读历史信息的优秀候选。
RAID 5	17-驱动器	94.1%	4（2 读—2 写）	减少了效率增益同时加倍了 RAID 设备的失效域。

随着条带中驱动器数的增加，RAID 5 效率随之增加。随着条带中磁盘数的增加，失效域（fault domain）也随之增加。条带中驱动器数的增加也增加了创建 RAID 设备所必需的最小磁盘数量。在一个实施例中，由于失效域大小的增加以及有限的效率增加，数据分级管理不使用大于 9 个驱动器的 RAID 5 条带大小。数据分级管理使用为快照页大小整数倍的 RAID 5 条带大小。这允许数据分级管理在将页移动

至 RAID 5 时执行全条带写，从而使得移动更有效。对数据分级管理的目的，所有的 RAID 5 配置具有相同的写 I/O 特征。例如，2.5 英寸 FC 磁盘上的 RAID 5 可能不能有效地使用这些磁盘的性能。为防止这种组合，数据分级管理需要支持防止 RAID 类型在某些磁盘类型上运行的能力。数据分级管理的配置也可防止系统使用 RAID 10 或 RAID 5 的空间。

下表中示出磁盘类型：

表 2：磁盘类型

类型	速度	成本	问题
2.5 英寸 FC	优	高	非常昂贵
FC 15 K PRM	良好	中	昂贵
FC 10 K PRM	良好	良好	合理价格
SATA	一般	低	便宜/较不可靠

数据分级管理包含对相对于系统内的驱动器的磁盘驱动器进行自动分类的能力。系统检查磁盘来确定它相对于系统中的其它磁盘的性能。较快速的磁盘被分类在较高值分类中，较慢磁盘被分类在较低值分类中。当磁盘被添加至系统时，系统自动重新平衡磁盘的值分类。该方法同时处理了从不改变的系统以及当添加新磁盘时经常改变的系统两者。自动分类可将多个磁盘类型置于同一值分类中。如果确定驱动器在值上足够接近，那么它们可具有相同的值。

在一个实施例中，系统包含以下驱动器：

高－10K FC 驱动器

低－SATA 驱动器

随着 15K FC 驱动器的添加，数据分级管理自动对磁盘重新分类，并降级该 10K FC 驱动器。这产生以下分类：

高－15K FC 驱动器

中－10K FC 驱动器

低－SATA 驱动器

在另一实施例中，系统可具有以下驱动器类型：

高－25K FC 驱动器

低－15K FC 驱动器

从而，该 15K FC 驱动器被分类为较低值分类，而 15K FC 驱动器被分类为较

高值分类。

如果将 SATA 驱动器添加至该系统，数据分级管理自动对磁盘重新分类。这产生以下分类：

高－25K FC 驱动器

中－15K FC 驱动器

低－SATA 驱动器

数据分级管理可包括瀑布式分级管理。通常，瀑布式分级管理仅当完全使用了资源时才将数据移动至较不昂贵的资源中。瀑布式分级管理有效地最大化最昂贵系统资源的使用。它也最小化系统的成本。向最低的池添加便宜的磁盘将在底部创建较大的池。

典型的瀑布式分级管理使用 RAID 10 空间，然后使用 RAID 空间中的下一个，诸如 RAID 5 空间。这迫使瀑布直接前进至下一类磁盘的 RAID 10。或者，数据分级管理可包括如图 24 中所示的混合 RAID 瀑布式分级管理。该替换数据分级管理方法解决了最大化磁盘空间和性能的问题，且允许存储转换成同一磁盘类中的更有效形式。该替换方法也支持 RAID 10 和 RAID 5 共享磁盘类的总资源的要求。这可要求配置 RAID 等级可对磁盘类使用的磁盘空间的固定百分比。从而，该替换的数据分级管理方法最大化昂贵存储的使用，同时允许对另一 RAID 类的空间共存。

该混合 RAID 瀑布式方法在存储受限时，也仅将页移动移动至较不昂贵的存储。诸如总磁盘空间的百分比等阈值限制某一 RAID 类型的存储量。这最大化系统中最昂贵存储的使用。当存储逼近其极限时，数据分级管理自动将页移动至较低成本的存储。数据分级管理为写峰值提供缓冲器。

可以理解，以上的瀑布式方法也将页立即移动至最低成本存储，因为在某些情况中，可能存在以及时的方式将历史和非可访问页移动至较不昂贵存储的需求。历史页也可被即时移动至较不昂贵存储。

图 18 示出了数据分级管理进程 1800 的流程图。数据分级管理对系统中的每一页连续检查其访问模式和存储成本以确定是否存在要移动的数据。数据分级管理也可确定存储是否达到其最大分配。

数据分级管理进程确定该页是否可由任何卷访问。该进程对 PITC 检查附加于历史的每一卷以确定是否引用该页。如果该页正被活动地使用，那么该页对升级或缓慢降级而言是合格的。如果该页不可由任何卷访问，那么将其移动至可用的最低成本存储。数据分级管理也将 PITC 期满之前的时间计算在内。如果快照调度 PITC

即将期满，那么没有页将分级管理。如果页池正以积极的模式操作，那么页可分级管理。

数据分级管理最近访问检测需要从对页的升级中消除活动的爆发。数据分级管理将读和写访问跟踪分离。这允许数据分级管理在可访问的 RAID 5 设备上保持数据。如病毒扫描或报告等操作仅读数据。如果存储短缺，则数据分级管理改变最近访问的资格鉴定。这允许数据分级管理更积极地将页降级。这也有助于当存储短缺时从下往上填充系统。

当系统资源变得短缺时，数据分级管理可积极地移动数据页。对于所有这些情况，仍然必须有更多的磁盘或配置改变。数据分级管理拉长了系统可在紧缺状况中操作的时间量。数据分级管理试图尽可能长时间地保持系统可操作。这一直持续到当它所有的存储类都耗尽空间时。

在当 RAID 10 空间短缺，且总的可用磁盘空间短缺的情况中，数据分级管理可调拨 RAID 10 磁盘空间来移动至更有效的 RAID 5 中。以写性能为代价，这增加了系统的总体容量。但仍必需有更多的磁盘。如果完全使用了特定的存储类，那么数据分级管理允许借用非可接受页以保持系统运行。例如，如果卷被配置成对其可访问信息使用 RAID 10-FC，那么它可以从 RAID 5-FC 或 RAID 10-SATA 中分配页，直到有更多的 RAID 10-FC 空间可用。

数据分级管理也支持压缩来增加系统的察觉的容量。压缩可仅用于不访问的历史页，或用作恢复信息的存储。压缩表现为接近存储成本底部的另一类存储。

如图 25 中所示，页池基本上包含自由列表和设备信息。页池需要支持多个自由列表、增强的页分配方案以及自由列表的分类。页池为每一类存储维护单独的自由列表。分配方案允许从多个池中的一个分配页，同时设定最小或最大允许的类。自由列表的分类来自设备配置。每一自由列表提供其自己的计数器用于统计汇集和显示。每一自由列表也提供 RAID 设备效率信息用于存储效率状态的汇集。

在一个实施例中，设备列表可要求跟踪存储类成本的额外能力。该组合确定存储的类。如果用户希望对所配置的类具有更多或更少粒度时，发生这一情况。

图 26 示出了高性能数据库的一个实施例，其中所有可用数据，即便最近未访问，也仅驻留在 2.5 FC 驱动器上。非可访问历史数据被移动至 RAID 5 光纤通道。

图 27 示出了 MRI 映像卷的一个实施例，其中对该动态卷而言可访问存储是 SATA RAID 10 和 RAID 5。如果映像最近未被访问，那么该映像被移动至 RAID 5。新的写最初进入 RAID 10。图 19 示出了经压缩的页布局的一个实施例。数据分级

管理通过对固定大小的数据页进行二次分配来实现压缩。二次分配信息跟踪该页的自由部分以及该页的已分配部分的位置。数据分级管理可以不预测压缩的效率，且可处理其二次分配内的可变大小页。

经压缩的页可显著地影响 CPU 性能。对写访问，经压缩的页将要求整个页被解压和重新压缩。从而，正被活动地访问的页不被压缩，并且返回至其未压缩状态。在存储极端受限的情况下，写可能是必需的。

PITC 重映射表指向二次分配信息，且被标记为指示被压缩的页。访问经压缩的页可比非压缩的页需要更高的 I/O 计数。访问可需要对二次分配信息的读取来检索实际数据的位置。该经压缩的数据可从磁盘中读取并可在处理器上解压。

数据分级管理可要求压缩能够对整个页的部分解压。这允许数据分级管理读访问仅解压页的小部分。读高速缓存的预读特征可有助于延迟压缩。单个解压可处理多个服务器 I/O。数据分级管理标记对压缩而言非良好候选的页，使得它将不必频繁尝试来压缩页。

图 20 示出了根据本发明的原理的高级磁盘驱动器系统中的数据分级管理的一个实施例。数据分级管理不改变卷的外部行为或数据路径的操作。数据分级管理可要求对页池的修改。页池基本上包含自由列表和设备信息。页池需要支持多个自由列表、增强的页分配方案以及自由列表的分类。页池为每一类存储维护单独的自由列表。该分配方案允许从多个池中的一个分配页，同时设定最小或最大允许的类。自由列表的分类可来自设备配置。每一自由列表提供其自己的计数器用于统计汇集和显示。每一自由列表也提供 RAID 设备效率信息用于存储效率统计的汇集。

PITC 标识用于移动的候选，并当移动可访问页时阻断对该页的 I/O。数据分级管理不断地对 PITC 检查候选。由于服务器 I/O、新快照页更新以及视图卷的创建/删除，页的可访问性不断改变。数据分级管理也不断检查卷配置改变，并概括页类和计数的当前列表。这允许数据分级管理评估该概述，并确定是否存在可能要移动的页。

每一 PITC 呈现用于每一类存储的页的数量的计数器。数据分级管理使用该信息来标识当达到阈值时成为移动页的良好候选的 PITC。

RAID 基于磁盘成本从一组磁盘中分配设备。RAID 也提供 API 来检索设备或潜在设备的效率。它也需要返回关于写操作所需的 I/O 数量的信息。数据分级管理也可要求 RAID NULL 使用第三方 RAID 控制器作为数据分级管理的一部分。RAID NULL 可消费整个磁盘，且可仅作为穿过的层。

磁盘管理器也可自动确定和存储磁盘分类。自动确定磁盘分类可要求对 SCSI 启动程序的改变。

通过以上描述和附图，本领域的普通技术人员可理解，所示和描述的具体实施例仅用于说明的目的，而不旨在限制本发明的范围。本领域的普通技术人员可以认识到，本发明也可用其它具体形式实现，而不背离本发明的精神或基本特征。对具体实施例的细节的参考不旨在限制本发明的范围。

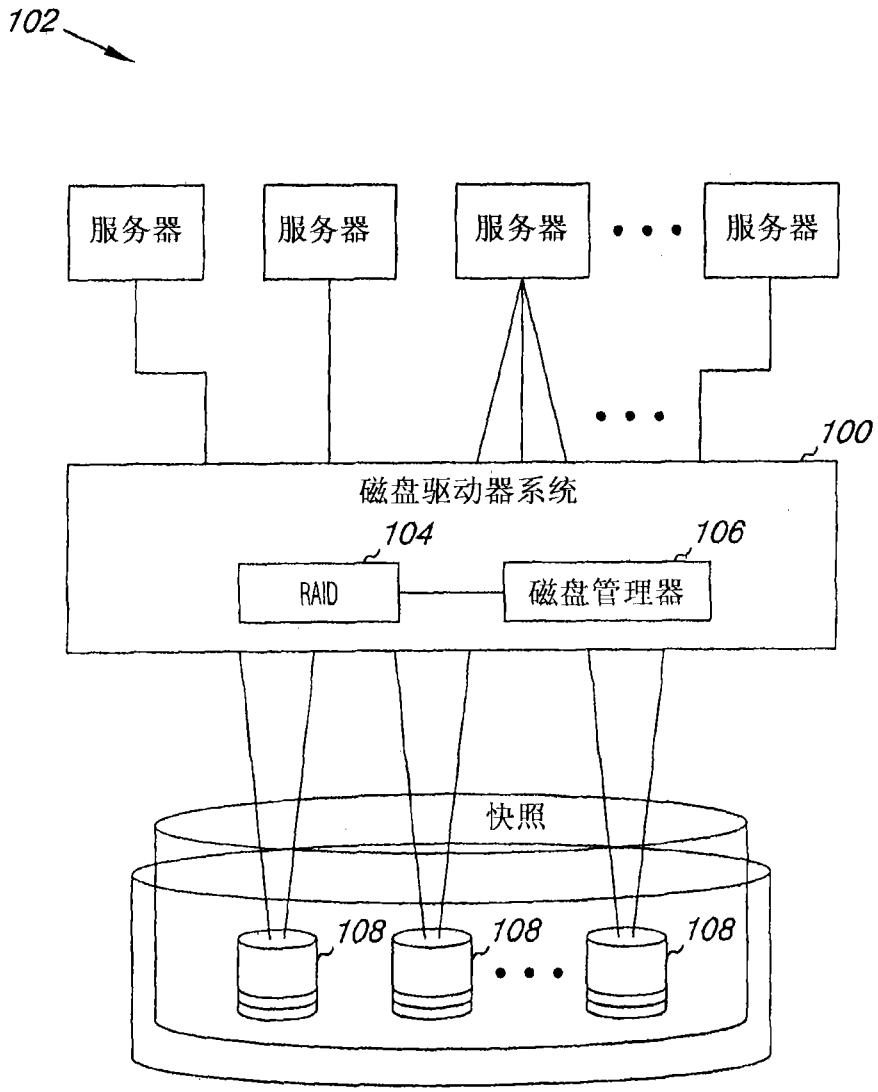

102
服务器
服务器
服务器
服务器
100
磁盘驱动器系统
104
106
RAID
磁盘管理器
快照
108
108
108

图 1

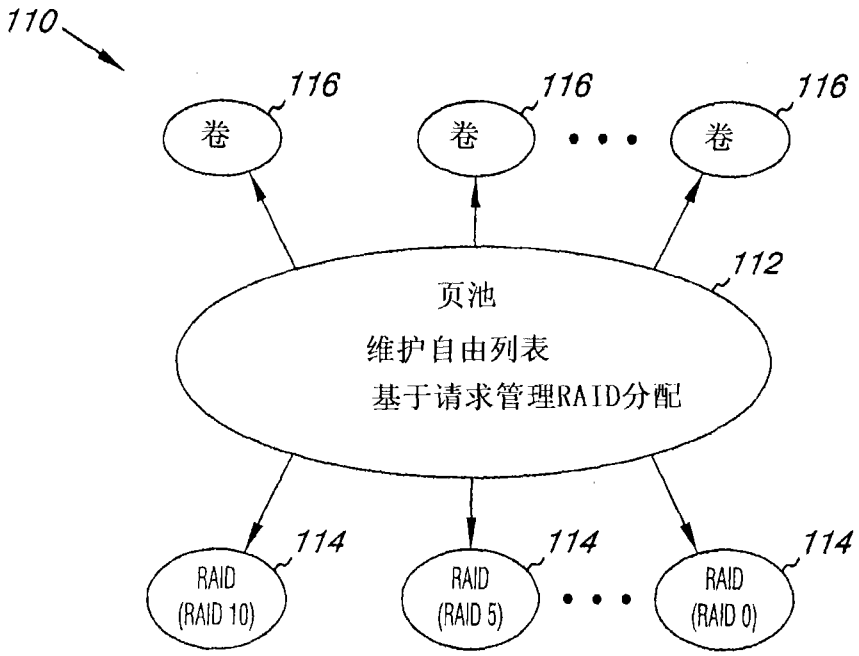
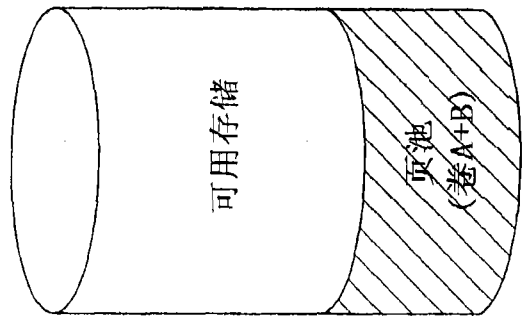

110
116
卷
116
卷
116
卷
112
页池
维护自由列表
基于请求管理RAID分配
114
RAID
(RAID 10)
114
RAID
(RAID 5)
114
RAID
(RAID 0)

图 2



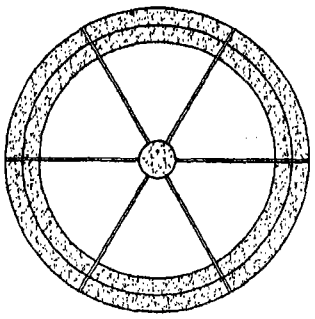
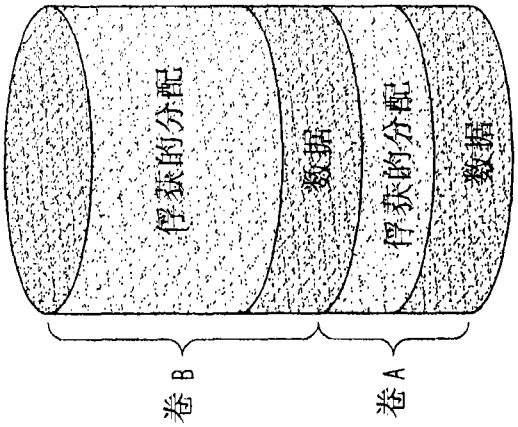


图 2B



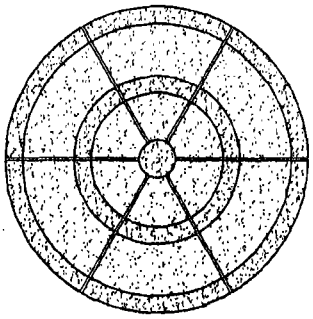


图 2A
(现有技术)

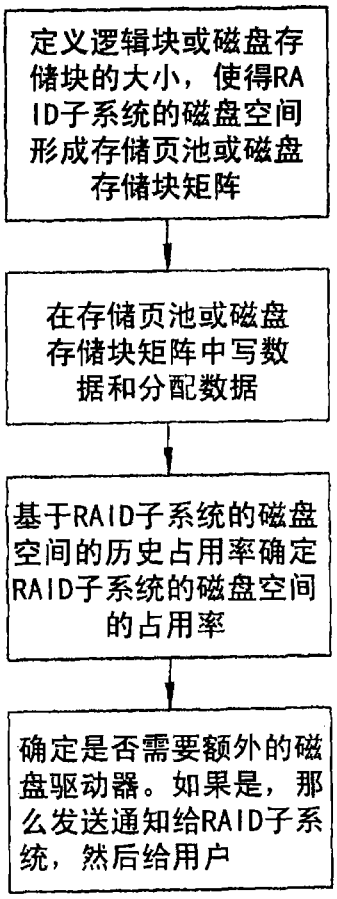

定义逻辑块或磁盘存储块的大小，使得RAID子系统的磁盘空间形成存储页池或磁盘存储块矩阵
在存储页池或磁盘存储块矩阵中写数据和分配数据
基于RAID子系统的磁盘空间的历史占用率确定RAID子系统的磁盘空间的占用率
确定是否需要额外的磁盘驱动器。如果是，那么发送通知给RAID子系统，然后给用户

图 2C

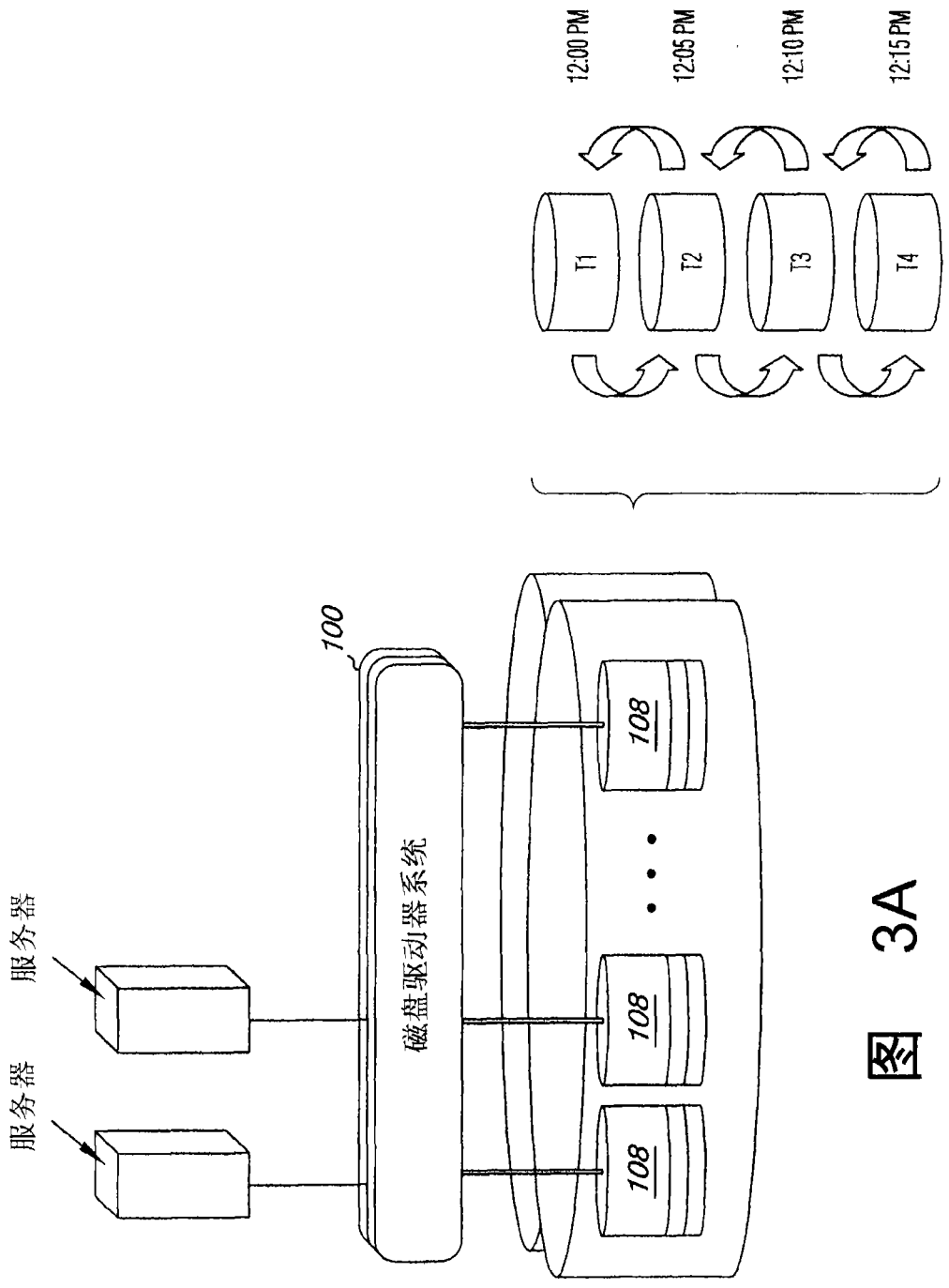

服务器
服务器
磁盘驱动器系统
100
108
108
108

图 3A

12:00 PM
12:05 PM
12:10 PM
12:15 PM
T1
T2
T3
T4

图 3B

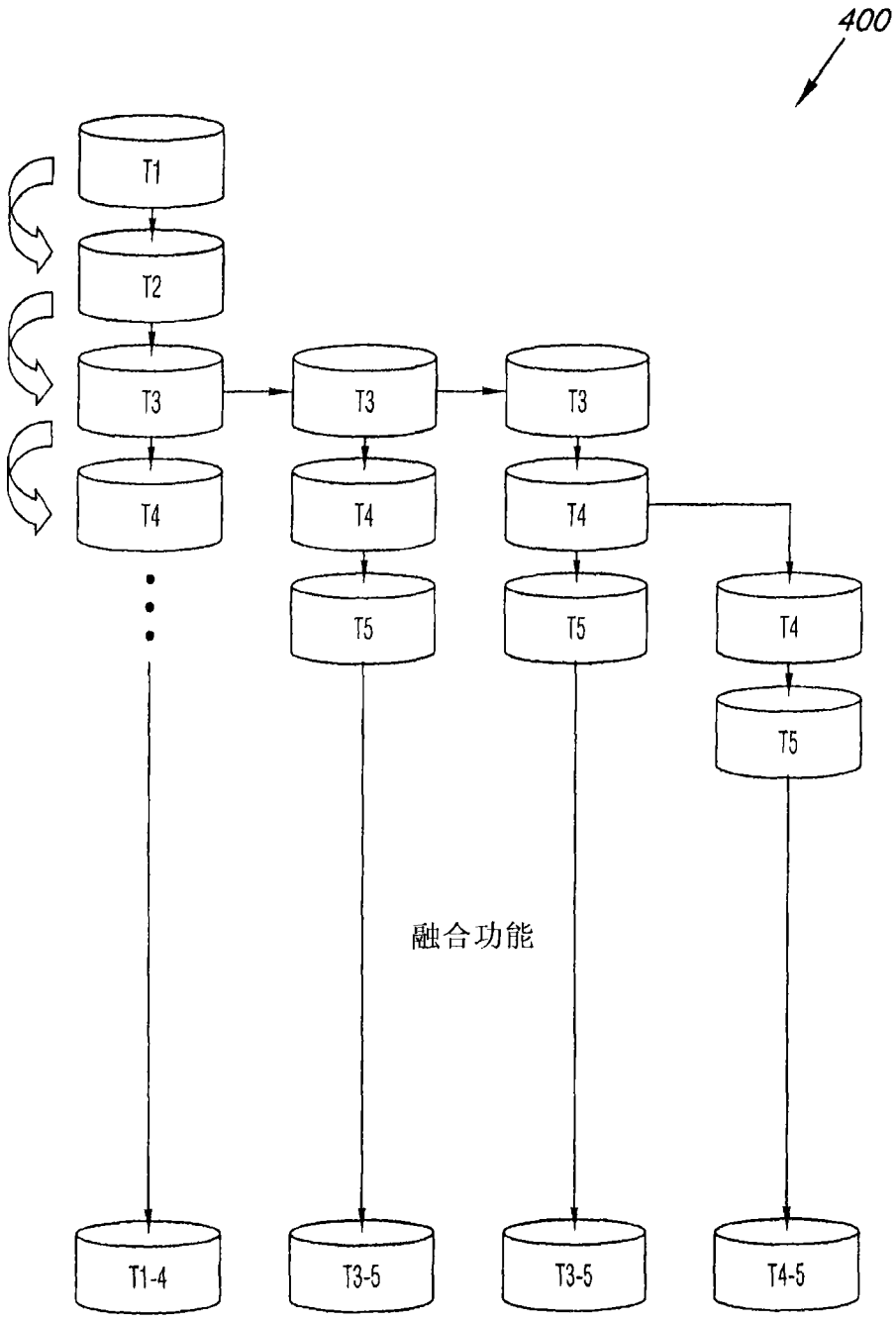
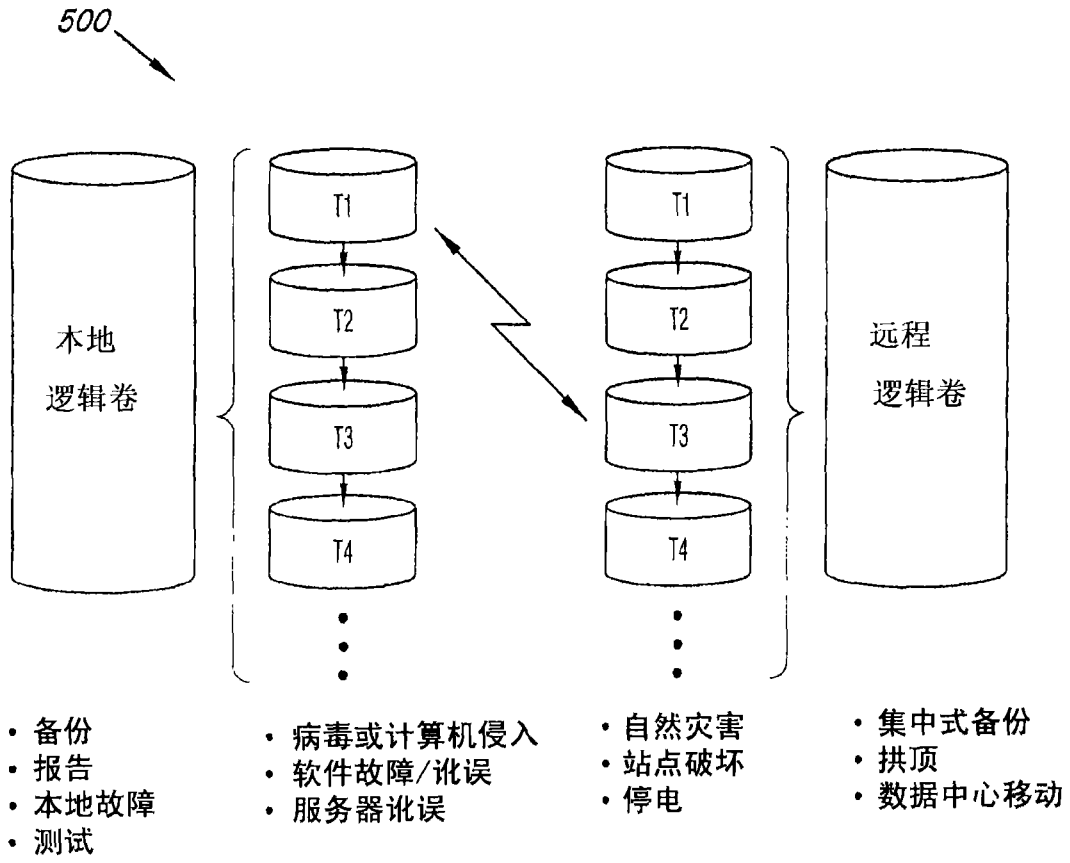


图 4


500
本地
逻辑卷
T1
T2
T3
T4
T1
T2
T3
T4
远程
逻辑卷
• 备份
• 报告
• 本地故障
• 测试
• 病毒或计算机侵入
• 软件故障/讹误
• 服务器讹误
• 自然灾害
• 站点破坏
• 停电
• 集中式备份
• 拱顶
• 数据中心移动

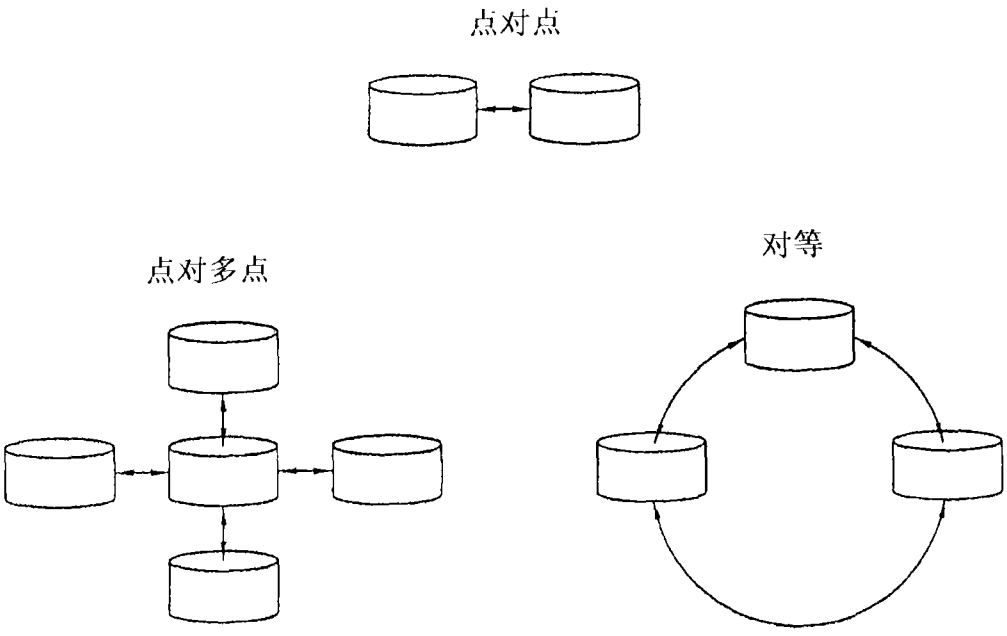

点对点
点对多点
对等

图 5

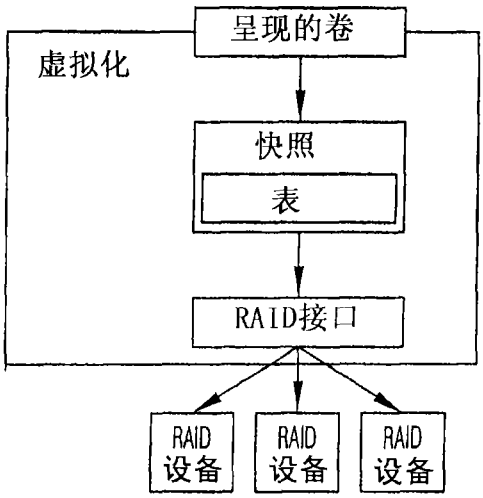


图 6

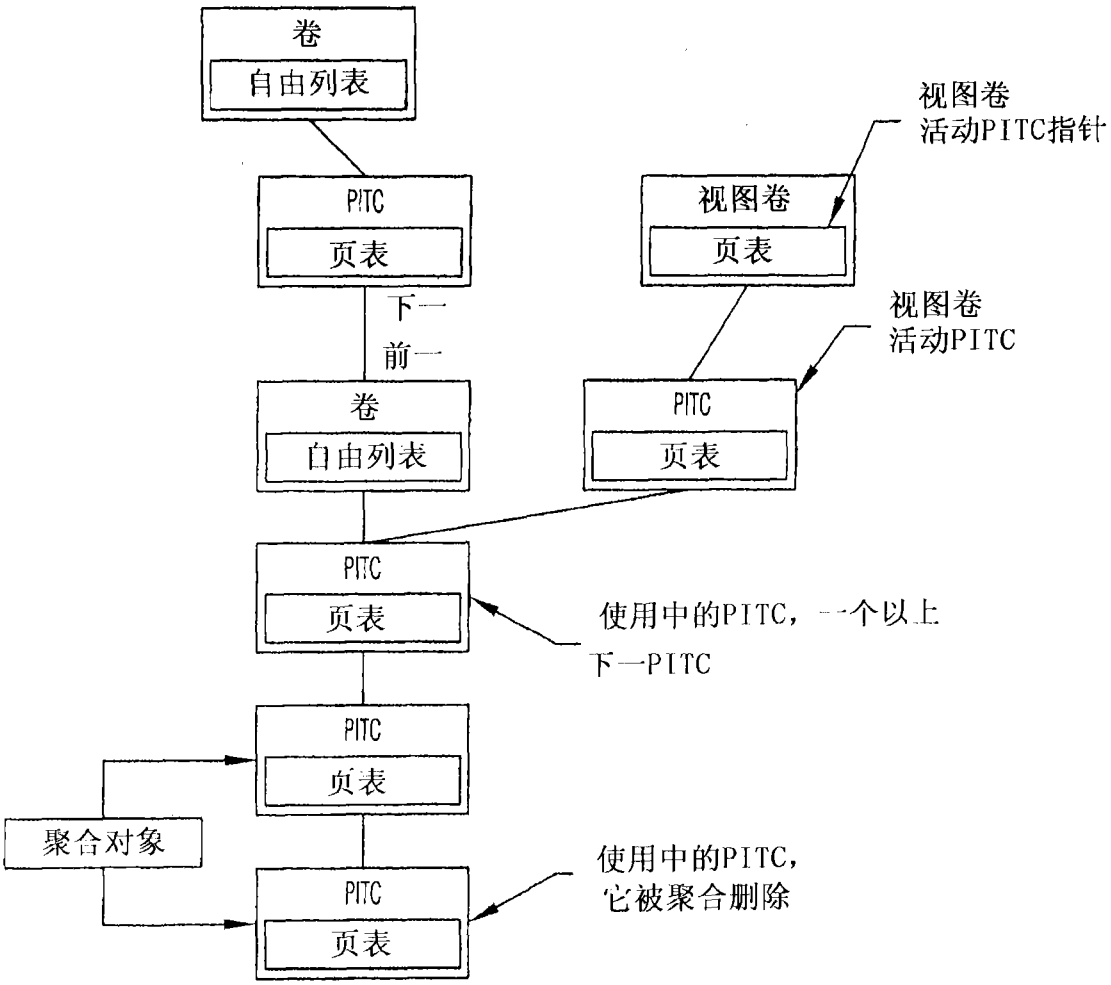

卷
自由列表
PITC
页表
下一
前一
卷
自由列表
视图卷
活动PITC指针
视图卷
页表
视图卷
活动PITC
PITC
页表
PITC
页表
使用中的PITC，一个以上
下一PITC
PITC
页表
聚合对象
PITC
页表
使用中的PITC，
它被聚合删除

图 7

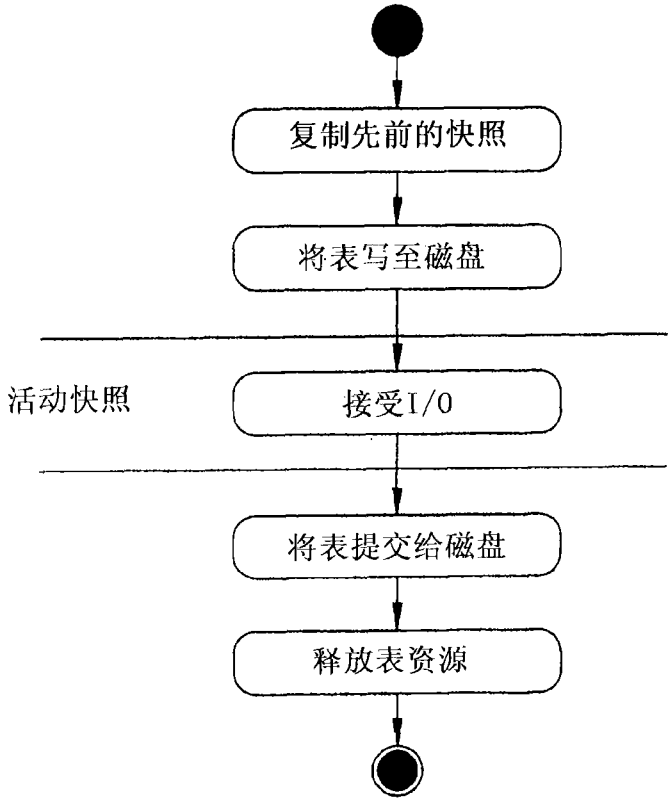


图 8

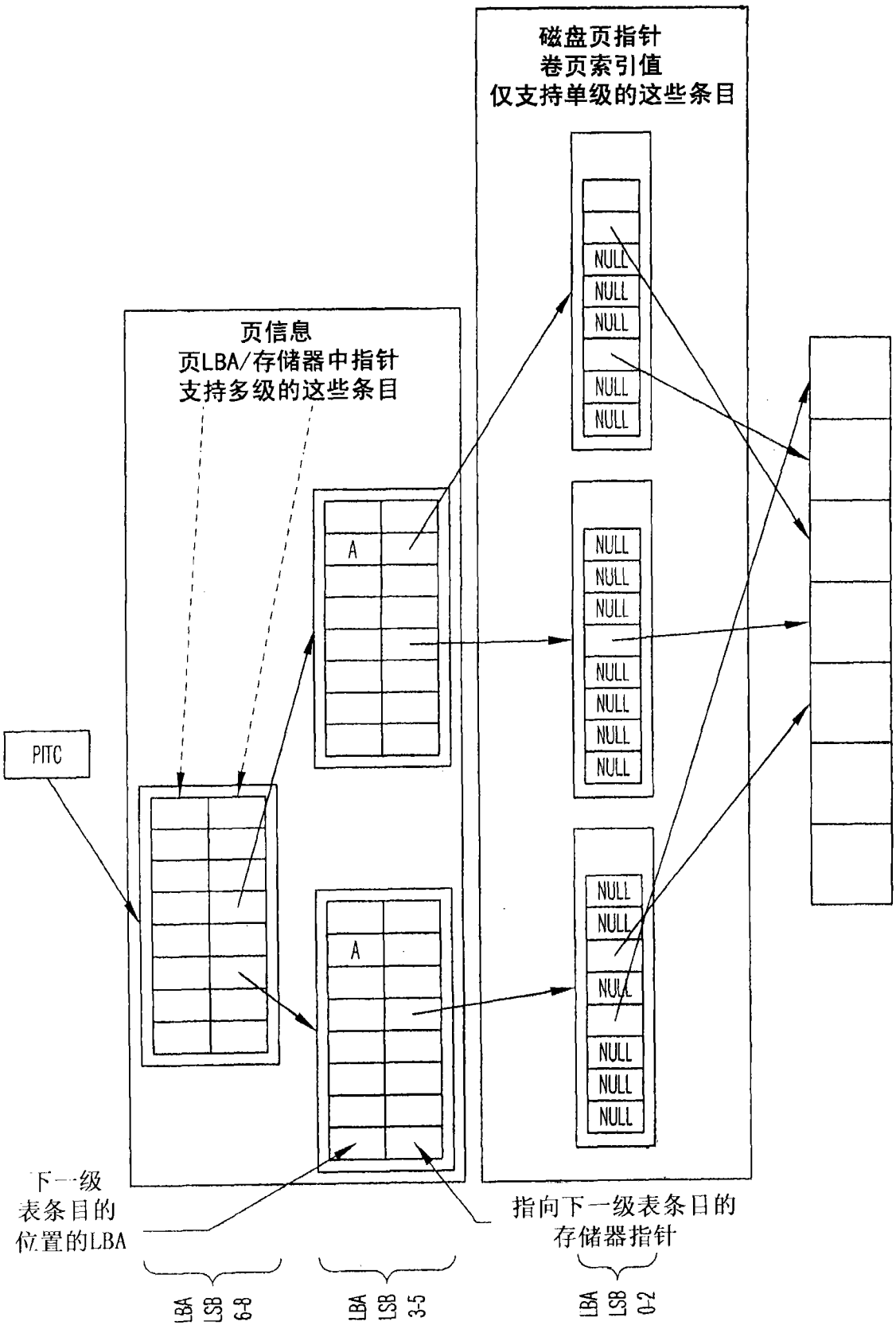

磁盘页指针
卷页索引值
仅支持单级的这些条目
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
NULL
页信息
页LBA/存储器中指针
支持多级的这些条目
A
A
PITC
下一级
表条目的
位置的LBA
指向下一级表条目的
存储器指针
LBA LSB 6-8
LBA LSB 3-5
LBA LSB 0-2

图 9

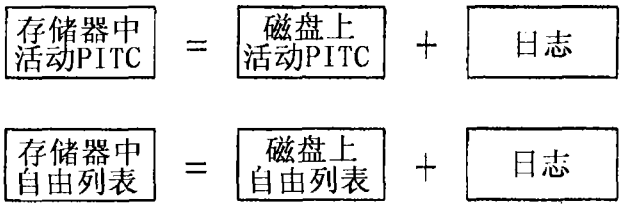

存储器中活动PITC
=
磁盘上活动PITC
+
日志
存储器中自由列表
=
磁盘上自由列表
+
日志

图 10

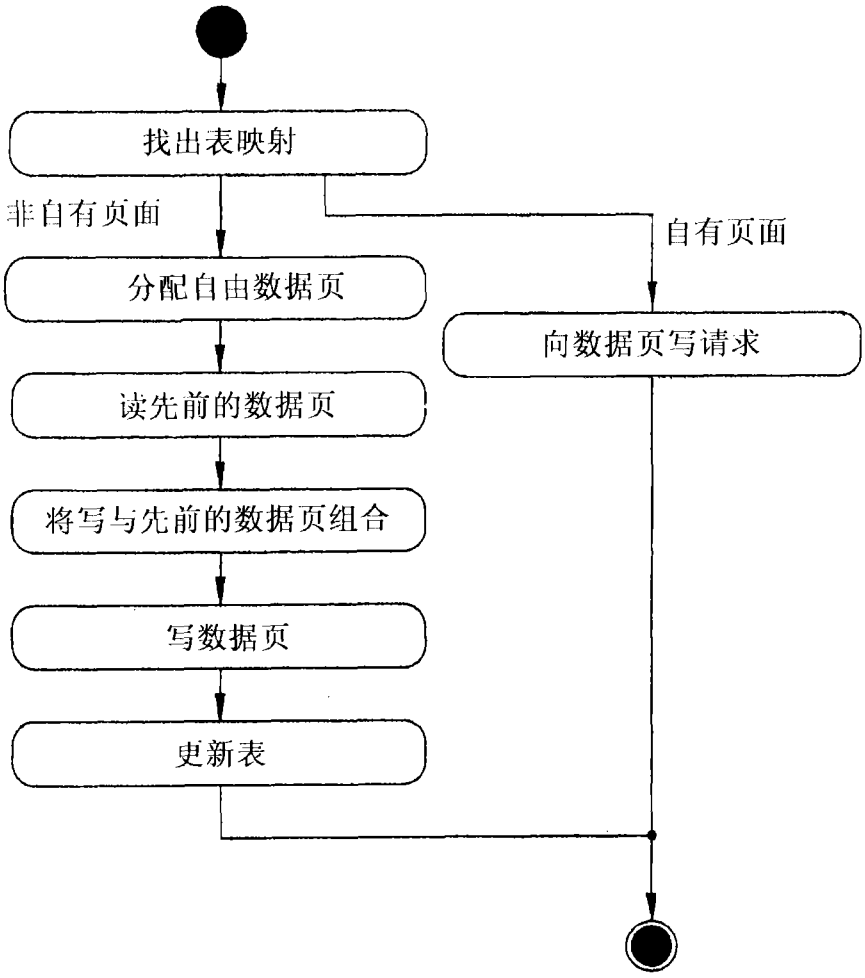

找出表映射
非自有页面
自有页面
分配自由数据页
向数据页写请求
读先前的数据页
将写与先前的数据页组合
写数据页
更新表

图 11

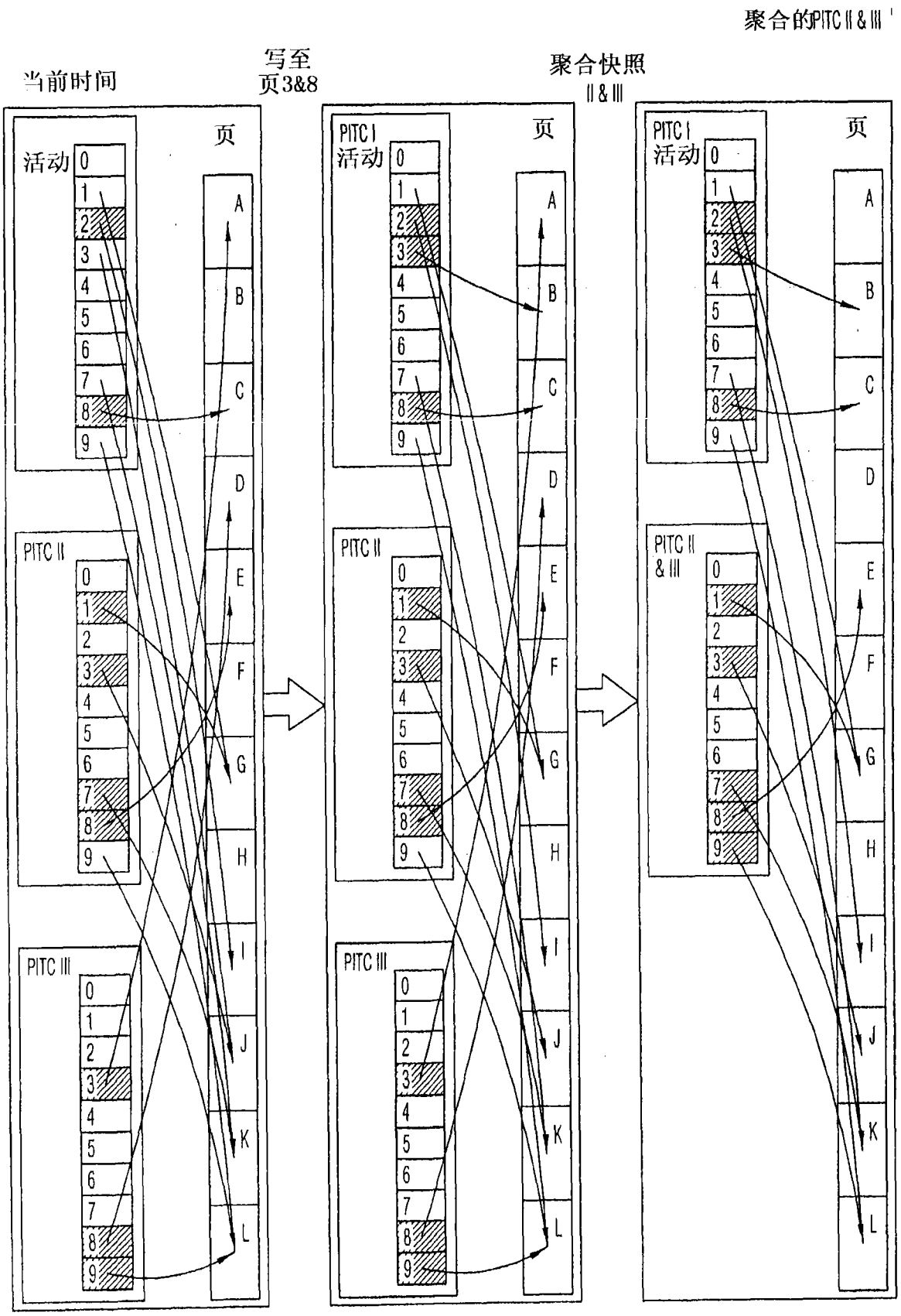

聚合的PITC II & III
当前时间
写至
页3&8
聚合快照
II & III
活动
页
PITC II
PITC III
PITC I
PITC II
& III
A
B
C
D
E
F
G
H
I
J
K
L

图 12

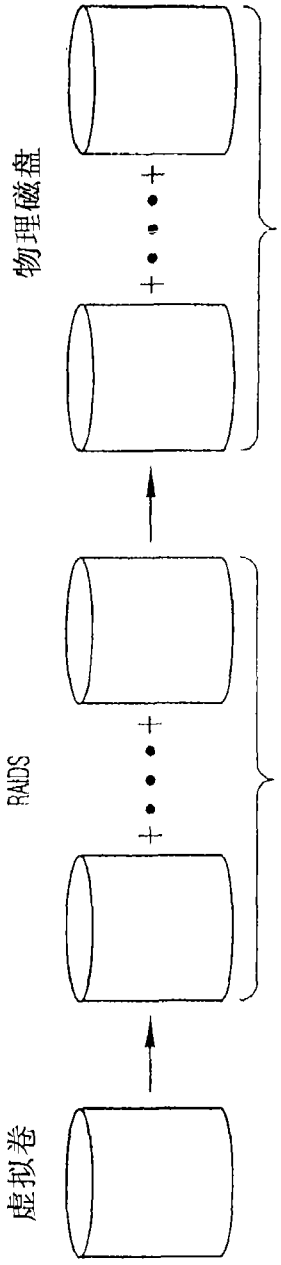


图 13A

（现有技术）

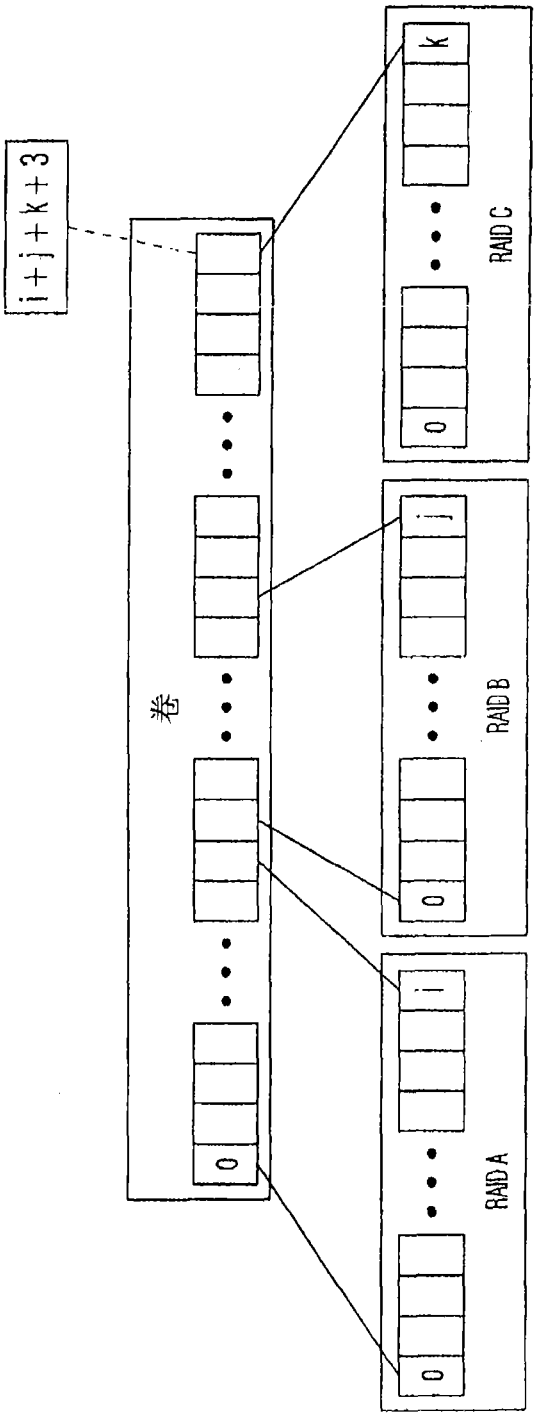


图 13B

（现有技术）

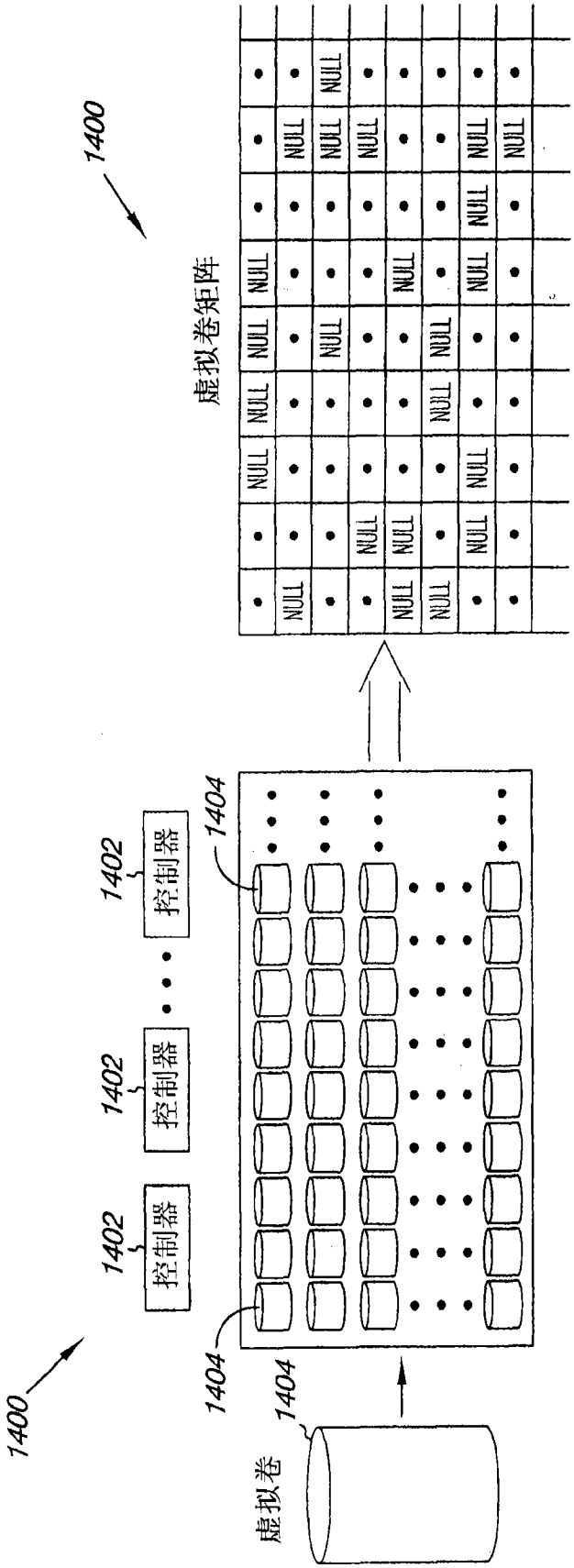

1400
虚拟卷矩阵
NULL
1400
1402
控制器
1402
控制器
1402
控制器
1404
1404
1404
虚拟卷

图 14A

图 14B

页 #	0	1	2	3	4	5	• • •
RAID设备	1	2	1	3	NULL	3	• • •
RAID 页 #	54	1	52	112	NULL	64	• • •

图 14C

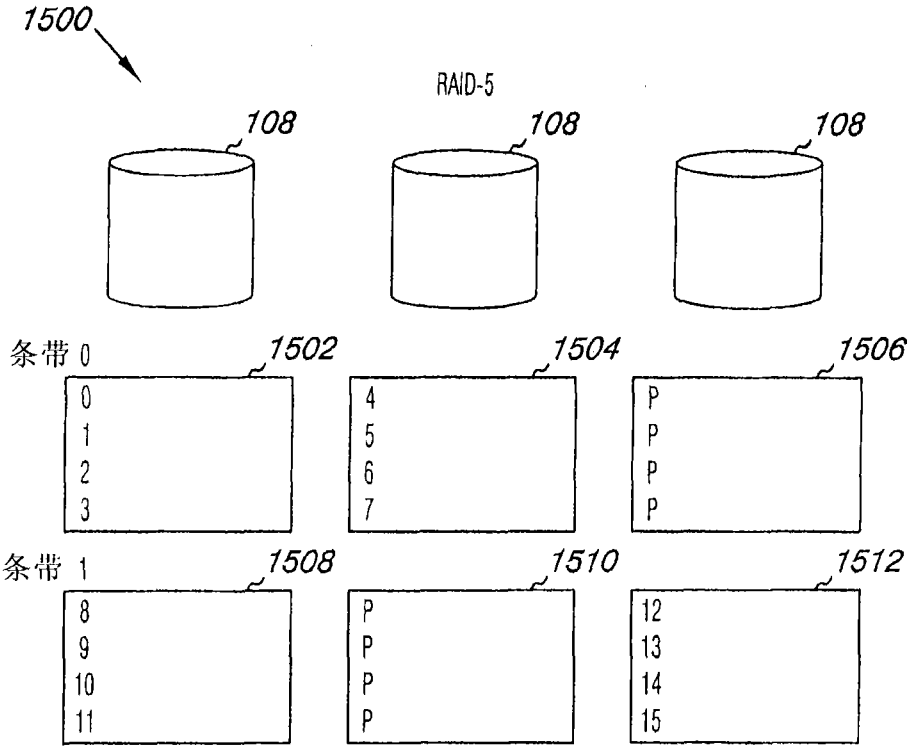


图 15

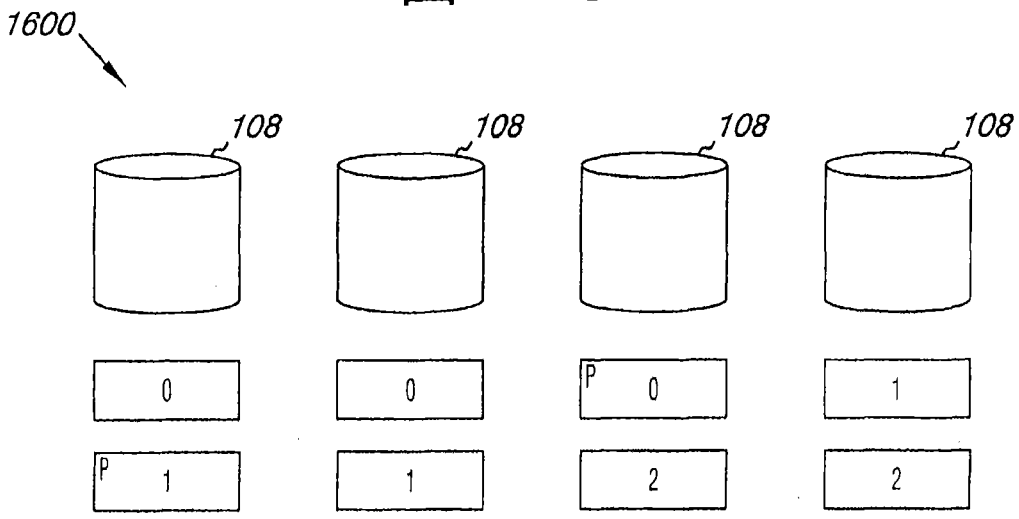


图 16

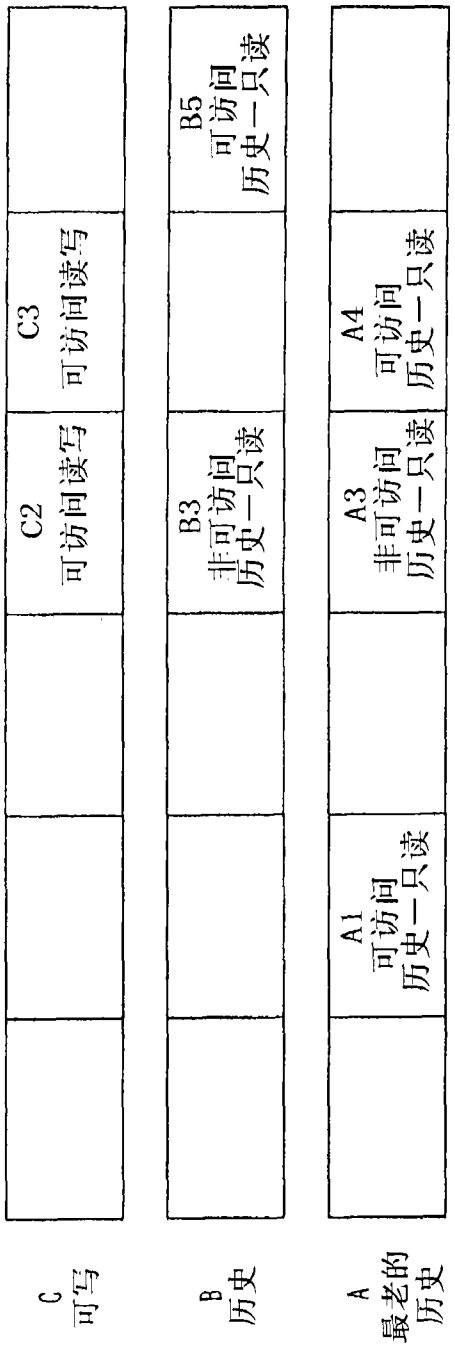

C
可写
B
历史
A
最老的
历史
C2
可访问读写
C3
可访问读写
B3
非可访问
历史—只读
B5
可访问
历史—只读
A1
可访问
历史—只读
A3
非可访问
历史—只读
A4
可访问
历史—只读

图 17

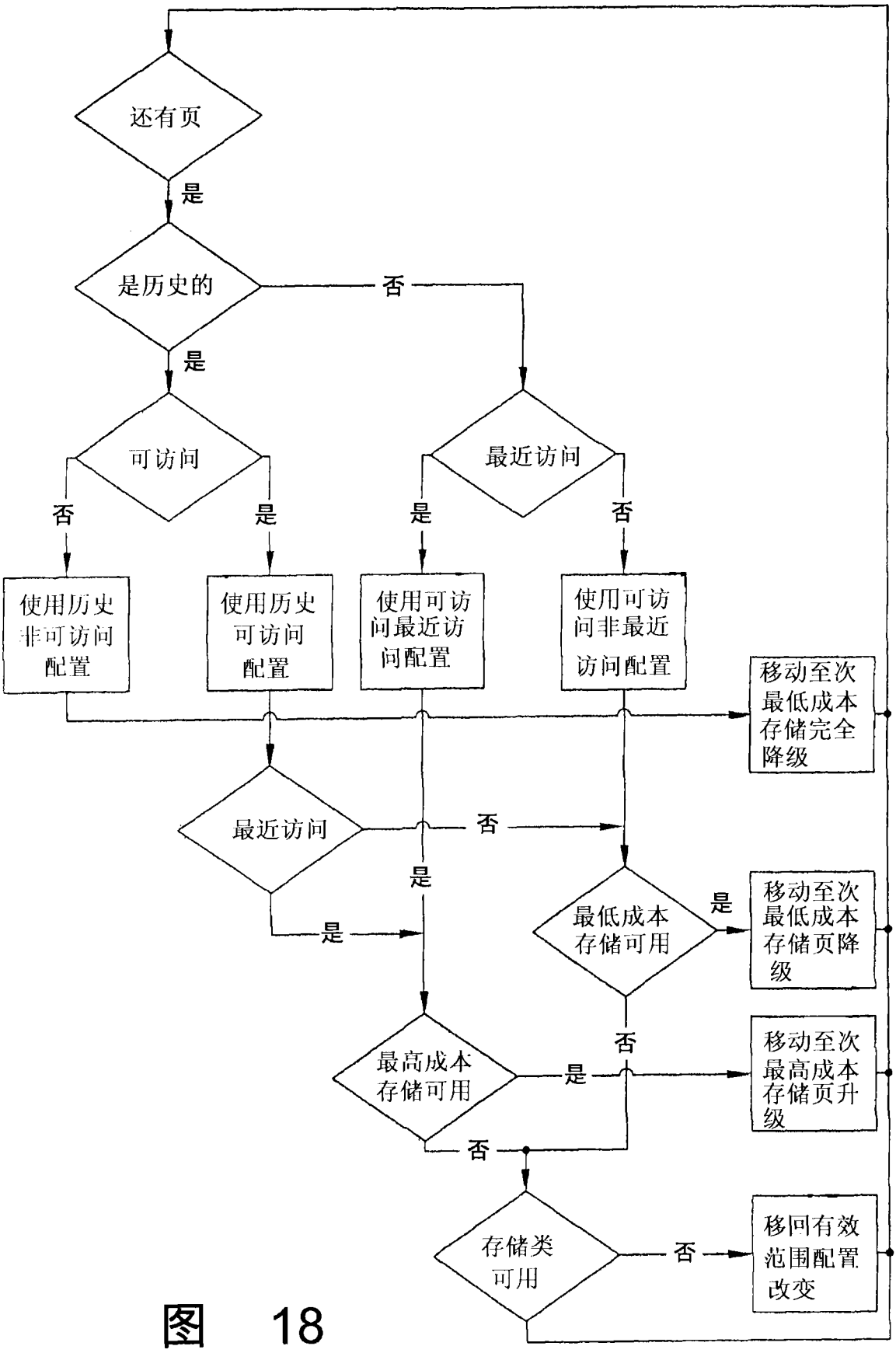

还有页
是
是历史的
否
是
可访问
最近访问
否
是
是
否
使用历史非可访问配置
使用历史可访问配置
使用可访问最近访问配置
使用可访问非最近访问配置
移动至次最低成本存储完全降级
最近访问
否
是
是
最低成本存储可用
是
移动至次最低成本存储页降级
否
最高成本存储可用
是
移动至次最高成本存储页升级
否
存储类可用
否
移回有效范围配置改变

图 18

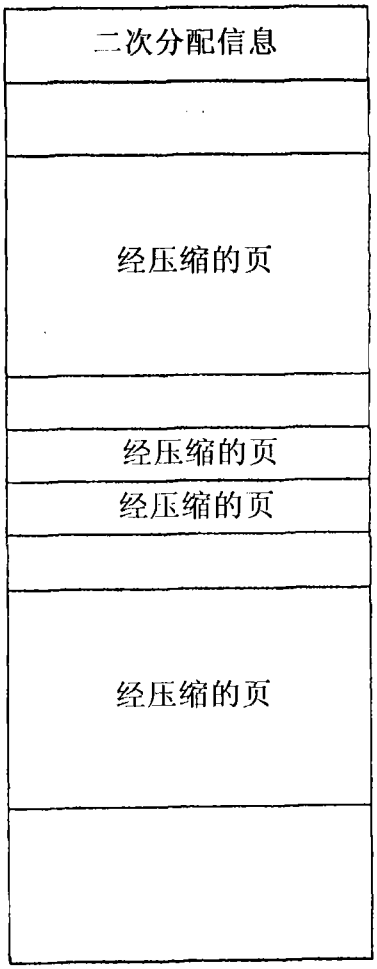


图 19

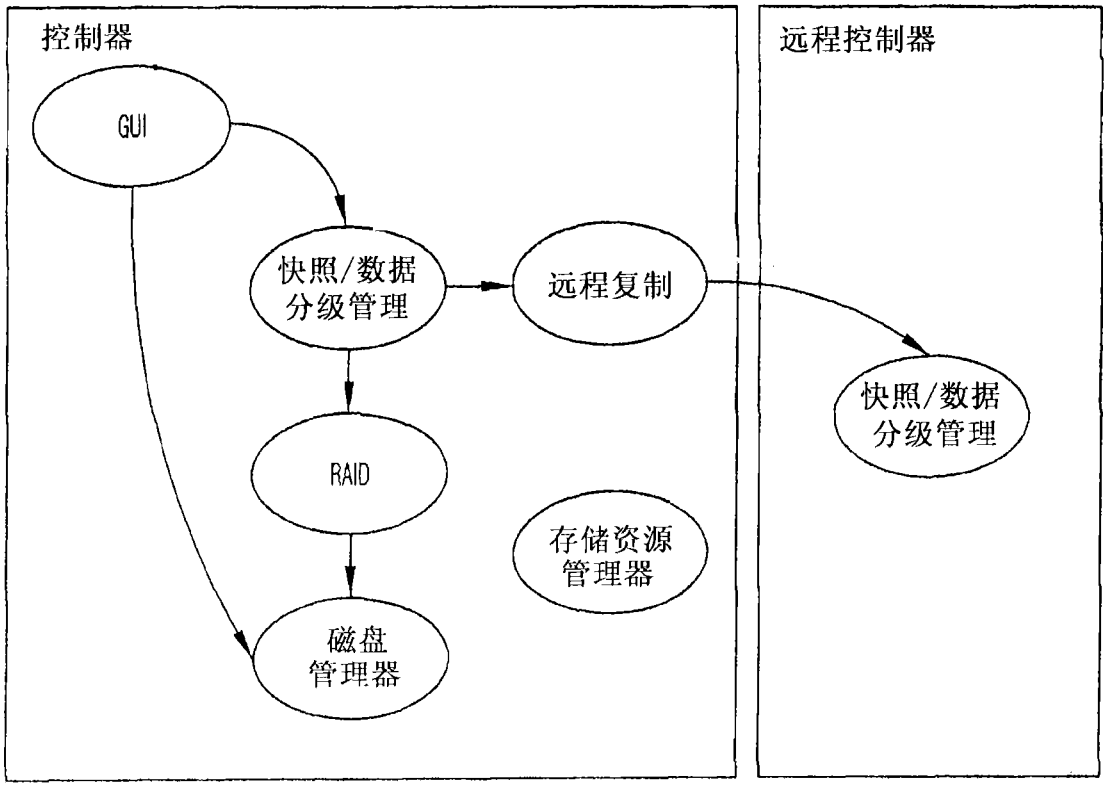

控制器
GUI
快照/数据
分级管理
远程复制
RAID
存储资源
管理器
磁盘
管理器
远程控制器
快照/数据
分级管理

图 20

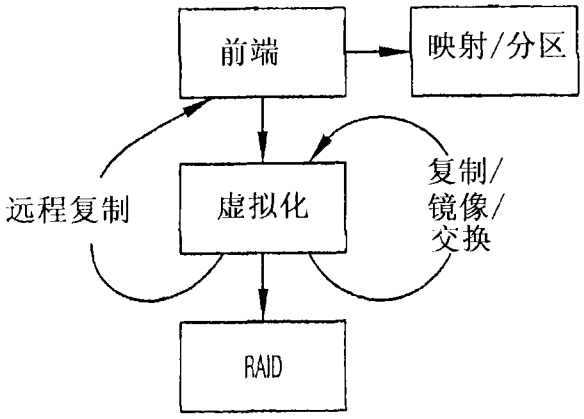

前端
映射/分区
远程复制
虚拟化
复制/
镜像/
交换
RAID

图 21

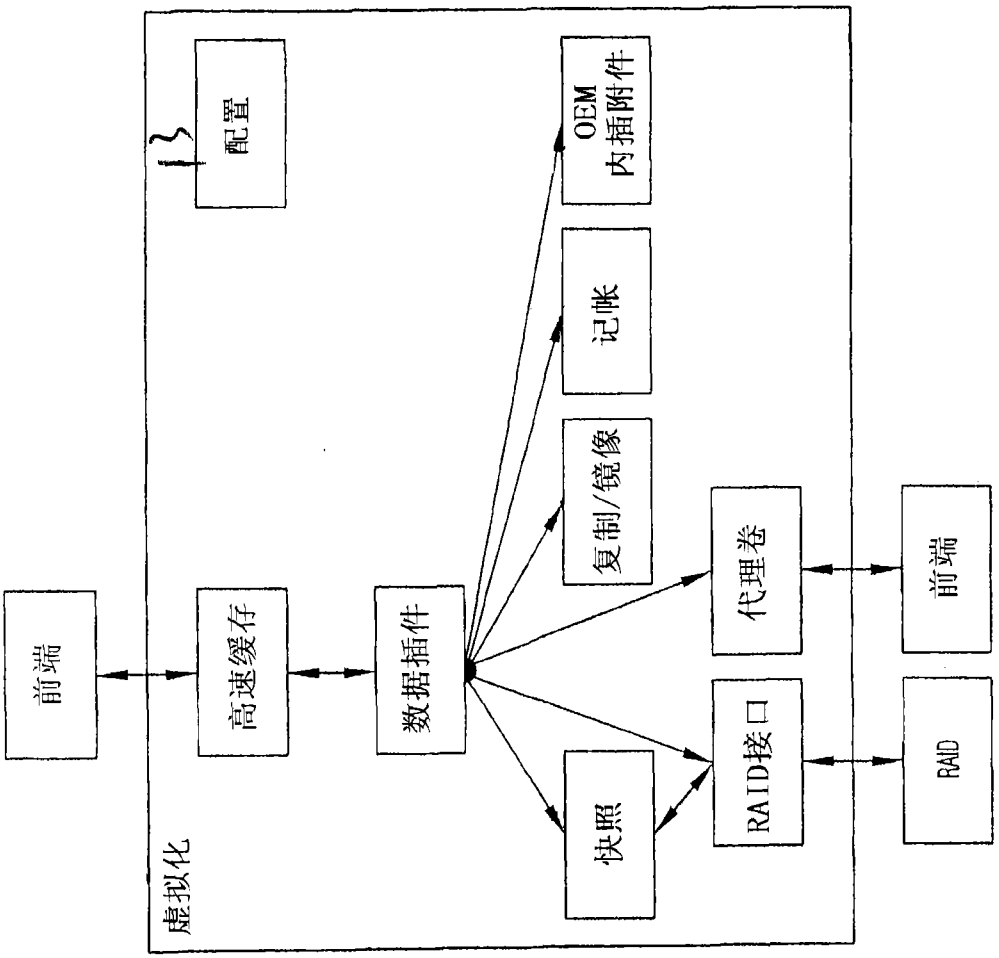


图 22

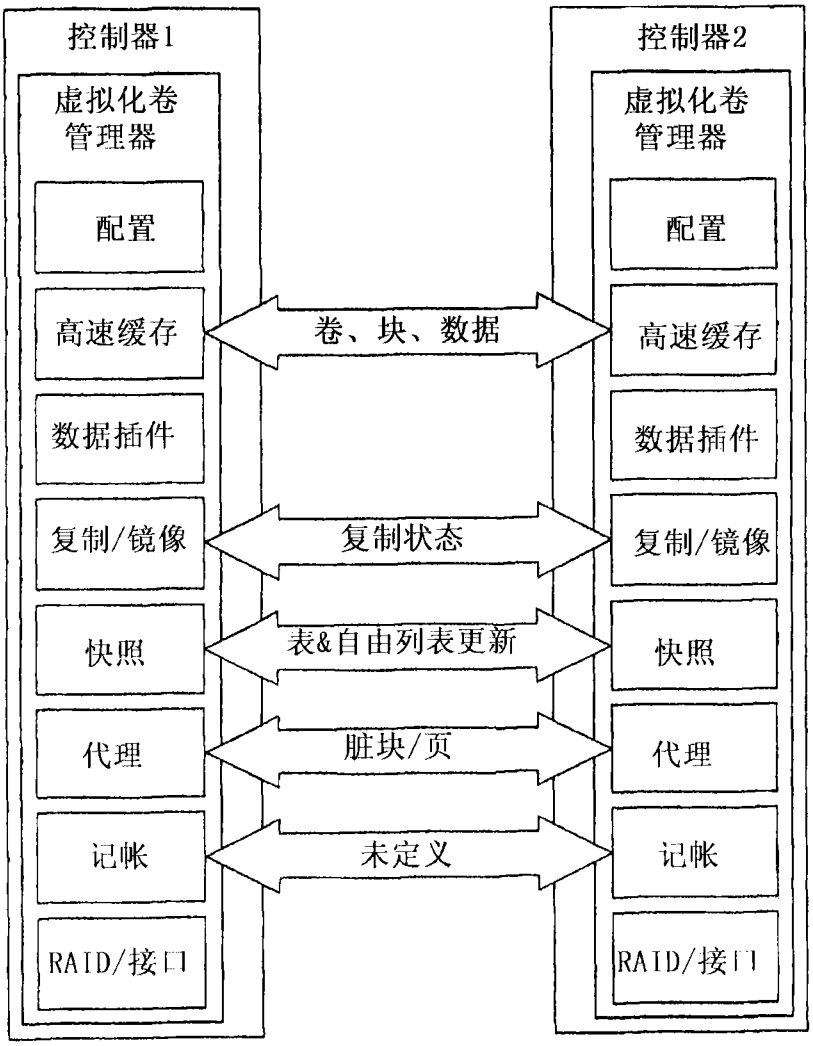

控制器1
虚拟化卷管理器
配置
高速缓存
数据插件
复制/镜像
快照
代理
记帐
RAID/接口
控制器2
虚拟化卷管理器
配置
高速缓存
数据插件
复制/镜像
快照
代理
记帐
RAID/接口
卷、块、数据
复制状态
表&自由列表更新
脏块/页
未定义

图 23

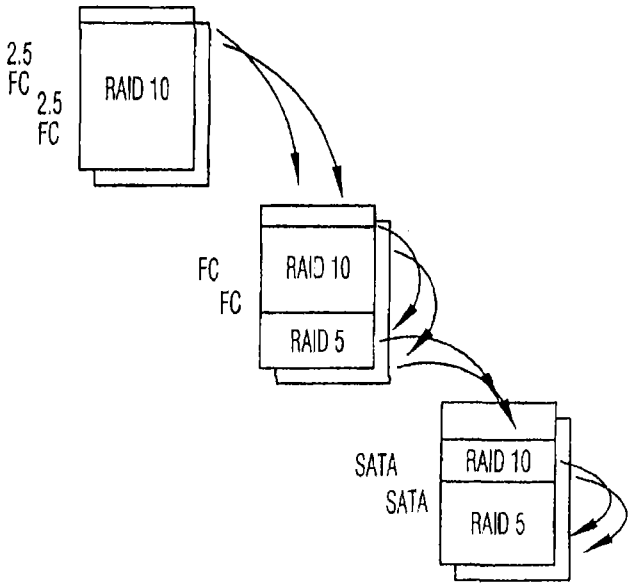

2.5
FC
2.5
FC
RAID 10
FC
FC
RAID 10
RAID 5
SATA
SATA
RAID 10
RAID 5

图 24

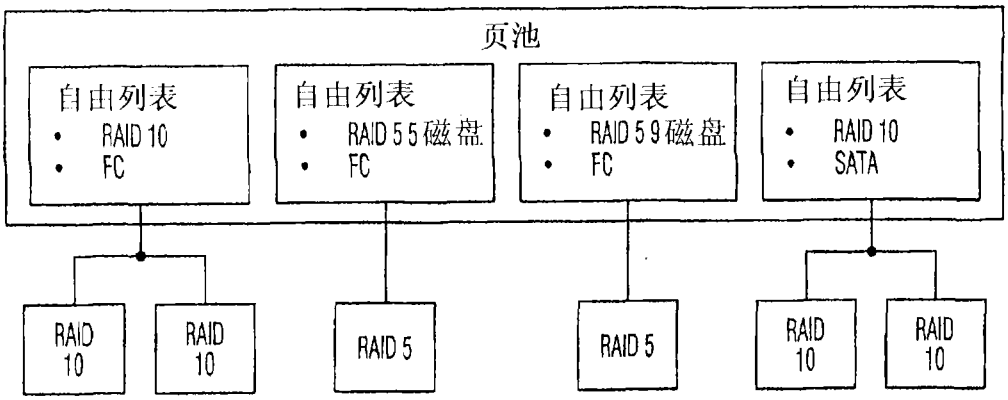

页池
自由列表
• RAID 10
• FC
自由列表
• RAID 5 5磁盘
• FC
自由列表
• RAID 5 9磁盘
• FC
自由列表
• RAID 10
• SATA
RAID 10
RAID 10
RAID 5
RAID 5
RAID 10
RAID 10

图 25

可访问历史

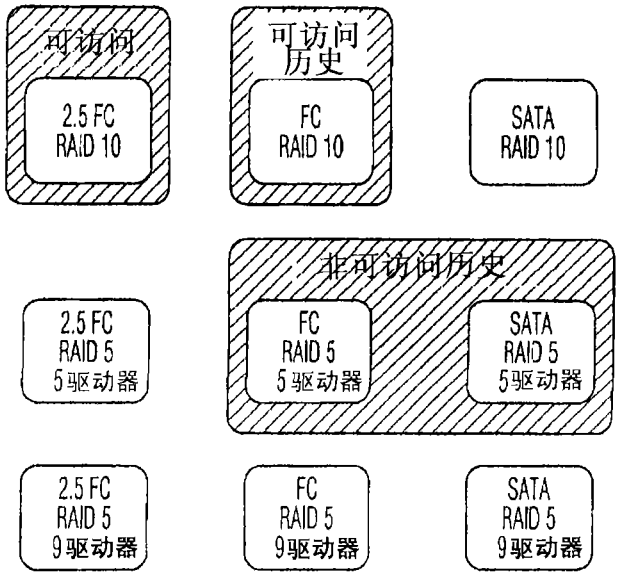


图 26

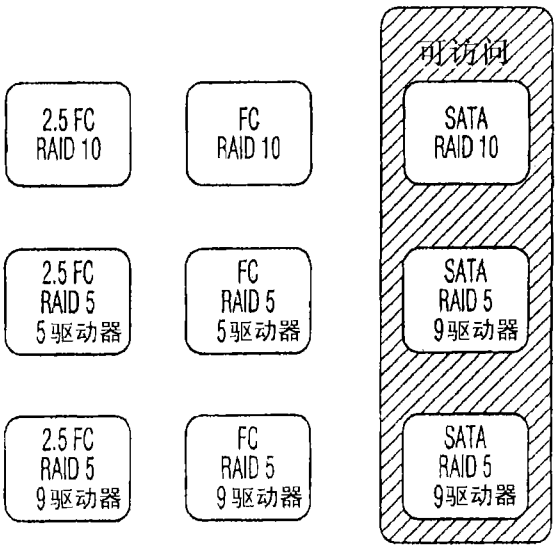


图 27