



(12) 发明专利申请

(10) 申请公布号 CN 102165448 A

(43) 申请公布日 2011.08.24

(21) 申请号 200980139428.1

(74) 专利代理机构 上海专利商标事务所有限公司 31100

(22) 申请日 2009.09.15

代理人 高见

(30) 优先权数据

12/241,912 2008.09.30 US

(51) Int. Cl.

G06F 17/40 (2006.01)

(85) PCT申请进入国家阶段日

G06F 17/30 (2006.01)

2011.03.29

(86) PCT申请的申请数据

PCT/US2009/057047 2009.09.15

(87) PCT申请的公布数据

W02010/039426 EN 2010.04.08

(71) 申请人 微软公司

地址 美国华盛顿州

(72) 发明人 M·K·斯里尼瓦斯 R·H·格伯

V·卡瑟瑞 J·F·路德曼

A·什日尼瓦斯 M·A·乌拉

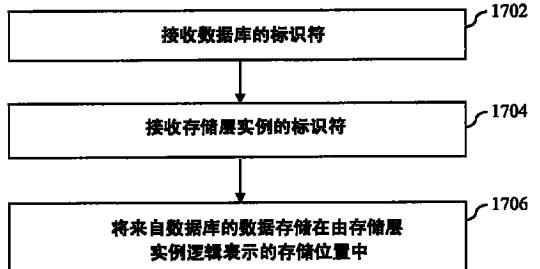
权利要求书 3 页 说明书 21 页 附图 13 页

(54) 发明名称

数据库服务器系统的存储层

(57) 摘要

描述了用于跨多个数据存储设备存储来自数据库的数据的技术，其中每一数据存储设备只能被一组互连的计算机系统中的对应的计算机系统访问。根据该技术，接收数据库的标识符。还接收存储层实例的标识符，其中该存储层实例包括每一个数据存储设备内的一个或多个存储位置的逻辑表示。响应于接收到数据库的标识符和存储层实例的标识符，将来自数据库的数据存储在由存储层实例逻辑表示的存储位置中的两个或更多个中，其中在其中存储数据的两个或更多个存储位置中的每一个在数据存储器中的对应的一个内。



1. 一种用于跨多个数据存储设备存储来自数据库的数据的方法,其中每一数据存储设备都只能够被一组互连的计算机系统中的对应的计算机系统访问,所述方法包括:

接收所述数据库的标识符(1702);

接收存储层实例的标识符,其中所述存储层实例包括所述数据存储设备的每一个内的一个或多个存储位置的逻辑表示(1704);以及

响应于接收到所述数据库的所述标识符和所述存储层实例的所述标识符,将来自所述数据库的数据存储在由所述存储层实例逻辑表示的所述存储位置中的两个或更多个中,其中在其中存储数据的所述两个或更多个存储位置中的每一个在所述数据存储设备中的对应的一个内(1706)。

2. 如权利要求1所述的方法,其特征在于,将来自所述数据库的数据存储在由所述存储层实例逻辑表示的所述存储位置中的两个或更多个中包括:

将来自所述数据库的数据的相同段的副本存储在所述两个或更多个存储位置里的每一个中。

3. 如权利要求1所述的方法,其特征在于,还包括创建所述存储层实例,其特征在于,创建所述存储层实例包括:

接收所述存储层实例的所述标识符;

接收所述数据存储设备中的每一个内的所述一个或多个存储位置中的每一个的标识符;以及

响应于接收到所述存储层实例的所述标识符以及所述数据存储设备中的每一个内的所述一个或多个存储位置中的每一个的所述标识符,将所述存储层实例与所述数据存储设备中的每一个内的所述一个或多个存储位置进行关联。

4. 如权利要求1所述的方法,其特征在于,还包括更改所述存储层实例,其特征在于,更改所述存储层实例包括:

接收所述存储层实例的所述标识符;

接收所述数据存储设备中的至少一个内未由所述存储层实例逻辑表示的至少一个存储位置的标识符;以及

响应于接收到所述存储层实例的所述标识符和未由所述存储层实例逻辑表示的所述至少一个存储位置的所述标识符,将所述至少一个存储区与所述存储层实例进行关联,以使得所述存储层实例逻辑地表示所述至少一个存储区。

5. 如权利要求4所述的方法,其特征在于,还包括:

响应于所述更改存储层实例,将来自所述数据库的数据存储在所述至少一个存储位置。

6. 如权利要求1所述的方法,其特征在于,还包括更改所述存储层实例,其特征在于,更改所述存储层实例包括:

接收所述存储层实例的所述标识符;

接收由所述存储层实例逻辑表示的至少一个存储位置的标识符;

响应于接收到所述存储层实例的所述标识符以及由所述存储层实例逻辑表示的所述至少一个存储位置的所述标识符,将所述至少一个存储位置与所述存储层实例解除关联,以使得所述存储层实例不再逻辑地表示所述至少一个存储位置。

7. 如权利要求 6 所述的方法,其特征在于,还包括:

响应于所述更改存储层实例,从所述至少一个存储位置移除来自所述数据库的数据。

8. 一种包括其中存储有控制逻辑的计算机可读介质的计算机程序产品,所述控制逻辑包括:

被安排成执行如权利要求 1-7 中的任一项所述的方法步骤的计算机可读取程序装置。

9. 一种系统,包括:

多个互连的计算机系统(202);

多个数据存储设备(204),所述数据存储设备中的每一个被连接到所述互连的计算机系统中的对应的一个(202),并且只能被其访问;以及

在所述互连的计算机系统中的至少一个上执行的计算机程序逻辑,所述计算机程序逻辑包括:

被配置成接收数据库的标识符并接收存储层实例的标识符的命令处理器(1112),其中所述存储层实例包括所述数据存储设备(204)的每一个内的一个或多个存储位置(1604)的逻辑表示;以及

数据虚拟化管理器(1612),所述数据虚拟化管理器(1612)被配置成响应于由所述命令处理器(1112)接收到所述数据库的所述标识符和所述存储层实例的所述标识符,将来自所述数据库的数据存储在由所述存储层实例逻辑表示的所述存储位置(1604)中的两个或更多个中,其中在其中存储数据的所述两个或更多存储位置(1604)中的每一个在所述数据存储设备(204)中的对应的一个内。

10. 如权利要求 9 所述的系统,其特征在于,所述命令处理器进一步被配置成接收所述存储层实例的所述标识符,接收所述数据存储设备中的每一个内的所述一个或更多个存储位置中的每一个的标识符,以及响应于接收到所述存储层实例的所述标识符和所述数据存储设备中的每一个内的所述一个或多个存储位置中的每一个的所述标识符,将所述存储层实例与所述数据存储设备中的每一个内的所述一个或多个存储位置进行关联。

11. 如权利要求 9 所述的系统,其特征在于,所述命令处理器进一步被配置成接收所述存储层实例的所述标识符,接收所述数据存储设备中的至少一个内未由所述存储层实例逻辑表示的至少一个存储位置的标识符,以及将所述至少一个存储位置与所述存储层实例进行关联,以使得响应于接收到所述存储层实例的所述标识符和未由所述存储层实例逻辑表示的所述至少一个存储位置的所述标识符,所述存储层实例逻辑地表示所述至少一个存储位置。

12. 如权利要求 11 所述的系统,其特征在于,所述数据虚拟化管理器进一步被配置成响应于所述至少一个存储位置与所述存储层实例的关联,将来自所述数据库的数据存储在所述至少一个存储位置。

13. 如权利要求 9 所述的系统,其特征在于,所述命令处理器进一步被配置成接收所述存储层实例的所述标识符,接收由所述存储层实例逻辑表示的至少一个存储位置的标识符,以及将所述至少一个存储位置与所述存储层实例解除关联,以使得响应于接收到所述存储层实例的所述标识符和由所述存储层实例逻辑表示的所述至少一个存储区的所述标识符,所述存储层实例不再逻辑地表示所述至少一个存储位置。

14. 如权利要求 13 所述的系统,其特征在于,所述数据虚拟化管理器进一步被配置成

响应于所述至少一个存储位置与所述存储层实例解除关联，从所述至少一个存储位置移除来自所述数据库的数据。

15. 如权利要求 9 所述的系统，其特征在于，所述数据虚拟化管理器被配置成通过将命令发送到在所述互连的计算机系统中的两个或更多个上执行的数据虚拟化管理器代理，将来自所述数据库的数据存储由所述存储层逻辑表示的所述存储位置中的两个或更多个中。

## 数据库服务器系统的存储层

[0001] 背景

[0002] 数据库服务器是被配置成向通常被称为“客户机”的计算机程序或计算机提供数据库服务。这样的数据库服务可包括,例如,将数据存储在数据库中,从数据库检索数据,修改存储在数据库中的数据,或执行与对存储在数据库中的数据的管理和利用有关的其他服务。为执行这些功能,数据库服务器可以被配置成对存储在数据库中的数据执行诸如搜索、排序,以及索引之类的功能。

[0003] 这样的服务器提供良好性能、高可用性,以及可缩放性是符合数据库服务器的管理员和用户的利益的。另外,这样的服务器应该提供使用、管理和管理的简便性。

[0004] 概述

[0005] 提供本概述是为了以精简的形式介绍将在以下详细描述中进一步描述的一些概念。本概述并不旨在标识出所要求保护的主题的关键特征或必要特征,也不旨在用于限定所要求保护的主题的范围。

[0006] 此处描述了用于跨多个数据存储设备存储来自数据库的数据的方法,其中每一数据存储设备只能够被一组互连的计算机系统中的对应的计算机系统访问。根据该方法,接收数据库的标识符。还接收存储层 (storage tier) 实例的标识符,其中该存储层实例包括每一个数据存储设备内的一个或多个存储位置的逻辑表示。响应于接收到数据库的标识符和存储层实例的标识符,将来自数据库的数据存储由存储层实例逻辑表示的存储位置中的两个或更多个中,其中在其中存储数据的两个或更多个存储位置中的每一个在数据存储设备中的对应的一个内。

[0007] 此处还描述了一种系统。该系统包括多个互连的计算机系统和多个数据存储设备。数据存储设备中的每一个连接到互连的计算机系统中的对应的一个,并只可被其访问。该系统还包括在互连的计算机系统中的至少一个上执行的计算机程序逻辑。计算机程序逻辑包括命令处理器和数据虚拟化管理器。命令处理器被配置成接收数据库的标识符,并接收存储层实例的标识符,其中存储层实例包括每一个数据存储设备内的一个或多个存储位置的逻辑表示。数据虚拟化管理器被配置成响应于命令处理器接收到数据库的标识符和存储层实例的标识符,将来自数据库的数据存储在由存储层实例逻辑表示的存储位置中的两个或更多个中,其中在其中存储数据的两个或更多个存储位置中的每一个在数据存储设备中的对应的一个内。

[0008] 此处还描述了一种计算机程序产品。该计算机程序产品包括在其上记录有用于允许处理单元跨多个数据存储设备存储来自数据库的数据的计算机程序逻辑的计算机可读介质,其中每一数据存储设备只能够被一组互连的计算机系统中的对应的计算机系统访问。计算机程序逻辑包括第一装置、第二装置和第三装置。第一装置用于使得处理单元能接收数据库的标识符。第二装置用于使得处理单元能接收存储层实例的标识符,其中存储层实例包括每一个数据存储设备内的一个或多个存储位置的逻辑表示。第三装置用于使得处理器响应于能接收到数据库的标识符和存储层实例的标识符,将来自数据库的数据存储在由存储层实例逻辑表示的存储位置中的两个或更多个中,其中在其中存储数据的两个或

更多个存储位置中的每一个在数据存储设备中的对应的一个内。

[0009] 下面将参考各个附图,详细描述本发明的进一步特点和优点,以及本发明的各实施例的结构和操作。值得注意的是,本发明不仅限于此处所描述的特定实施例。这样的实施例只是出于例示的目的。基于此处所包含的原理,另外的实施例对那些相关领域技术人员是显而易见的。

[0010] 附图简述

[0011] 结合到本说明书并构成本说明书的一部分的附图示出了本发明,且与描述一起,进一步用于说明本发明的原理,并允许那些精通相关技术人员实施和使用本发明。

[0012] 图 1 是其中可以实现本发明的实施例的示例数据库系统的框图。

[0013] 图 2 是一数据库系统的框图,其中在单个计算机系统上安装并执行包括数据库服务器的实例和群集基础结构逻辑的对应的实例的砖 (brick)。

[0014] 图 3 是其中在同一个计算机系统上安装并执行两个或更多砖的数据库系统的框图。

[0015] 图 4 是包括多个存储位置的数据存储设备的框图。

[0016] 图 5 是示出了群集基础结构逻辑的代表性实例的框图。

[0017] 图 6 是示出了可以被包括在群集基础结构逻辑的实例内的一个或多个管理器的框图。

[0018] 图 7 是示出了被包括在群集基础结构逻辑的实例内的多个代理的框图。

[0019] 图 8 是示出了一个表和从其中导出的分区之间的关系的示图。

[0020] 图 9 是示出了一个分区和从其中导出的段之间的关系的示图。

[0021] 图 10 是一数据库系统的框图,该数据库系统包括段的物理表示的克隆 (clone) 跨与不同的计算机系统相关联的数据存储设备分布。

[0022] 图 11 是描绘了可能涉及执行涉及存储层实例的创建、更改或丢弃的功能的实体的框图。

[0023] 图 12 描绘了可以用来创建存储层的示例方法的流程图。

[0024] 图 13 描绘了可以用来更改现有存储层实例以将一个或多个新存储位置与存储层实例相关联的示例方法的流程图。

[0025] 图 14 描绘了可以用来更改现有存储层实例以将一个或多个新存储位置与存储层实例解除关联的示例方法的流程图。

[0026] 图 15 描绘了可以用来丢弃现有存储层实例的示例方法的流程图。

[0027] 图 16 是描绘了在执行与将数据库指派到存储层实例以及据此存储来自数据库的数据有关的功能时可能涉及的实体的框图。

[0028] 图 17 描绘了可以用来将数据库与存储层实例相关联并可以据此存储来自数据库的数据的方法的流程图。

[0029] 图 18 描绘了可以被用来实现本发明的各个方面示例基于处理器的计算机系统。

[0030] 通过下面的结合附图对本发明进行的详细说明,本发明的特点和优点将变得更加显而易见,在附图形中,类似的附图标记在整个说明书中标识对应的元素。在附图中,相同的附图标记一般指示相同的、功能上类似的和 / 或在结构上类似的元素。元素首先在其中

出现的附图由对应的附图标记中最左边的数字来指示。

[0031] 详细描述

[0032] A. 示例操作环境

[0033] 图 1 是其中可以实现本发明的实施例的示例数据库系统 100 的框图。如图 1 所示,系统 100 包括多个砖,标示为砖  $102_1, 102_2, 102_3, \dots, 102_n$ , 其中  $n$  指示系统 100 中砖的总数。每个砖包括数据库服务器 112 的实例以及与其通信地耦合的群集基础结构逻辑 114 的实例。具体而言,砖  $102_1$  包括数据库服务器  $112_1$  的实例以及与其通信地耦合的群集基础结构逻辑  $114_1$  的实例,砖  $102_2$  包括数据库服务器  $112_2$  的实例以及与其通信地耦合的群集基础结构逻辑  $114_2$  的实例,等等。虽然系统 100 被示为包括三个以上的砖,但是,可以理解,系统 100 也可以包括仅两个砖或仅三个砖。如在图 1 中进一步示出的,每一砖  $102_1-102_n$  通过通信基础结构 104 连接到每一个其他砖  $102_1-102_n$ 。

[0034] 每一数据库服务器实例  $112_1-112_n$  包括被配置成向此处被称为“客户机”的其他计算机程序或计算机提供数据库服务的计算机程序的实例。这样的数据库服务可包括,例如,将数据存储在数据库中,从数据库检索数据,修改存储在数据库中的数据,或执行与对存储在数据库中的数据的管理和利用有关的其他服务。为执行这些功能,每一数据库服务器实例  $112_1-112_n$  可以被配置成对存储在数据库中的数据执行诸如搜索、排序,以及索引之类的功能。在一个实施例中,数据库服务器  $112_1-112_n$  的每一实例包括由华盛顿州雷德蒙市的微软公司发布的 Microsoft® SQL Server® 的一个版本,虽然本发明没有这样的限制。

[0035] 群集基础结构逻辑  $114_1-114_n$  的每一实例包括计算机程序逻辑,该计算机程序逻辑被配置成允许多个数据库服务器实例  $112_1-112_n$  作为单个逻辑数据库系统一起操作,以便向与数据库服务器实例  $112_1-112_n$  进行交互的每一用户 / 客户机呈现单个系统映像。群集基础结构逻辑  $114_1-114_n$  的每一实例还被配置成允许与单个数据库相关联的数据被多个数据库服务器实例  $112_1-112_n$  同时存储、检索、修改或以其他方式处理。

[0036] 在数据库系统 100 的一种实现中,图 1 中所示的数据库服务器实例  $112_1-112_n$  和对应的群集基础结构实例  $114_1-114_n$  的每一组合被安装在基于对应的处理器的计算机系统上,并在其上执行,以执行如前所述的功能以及其他功能。在本文中的别处参考图 18 描述了一个这样的基于处理器的计算机系统的示例。

[0037] 例如,图 2 是数据库系统 100 的一种实现的框图,其中在基于单个处理器的计算机系统 202 上安装和执行包括数据库服务器实例  $112_1$  和群集基础结构逻辑实例  $114_1$  的砖  $102_1$ 。如图 2 所示,计算机系统 202 连接到通信基础结构 104,并连接到一个或多个数据存储设备 204。在一种实现中,数据存储设备 204 只能被计算机系统 202 访问。在这样的实现中,由数据库系统 100 的上下文内的数据库服务器实例  $112_1$  存储、检索、修改或以其他方式处理的任何数据库数据将被存储在附连到计算机系统 202 的数据存储设备 204 上。

[0038] 数据存储设备 204 可包括任何类型的直接附连存储器 (DAS) 设备,包括,但不仅限于硬盘驱动器、光驱动器,或可以通过诸如串行高级技术附连 (SATA) 接口、小型计算机系统接口 (SCSI)、串行附连 SCSI (SAS) 接口或光纤通道接口之类的标准接口直接地附连到计算机系统 202 的其他类型的驱动器。数据存储设备 204 也可以包括可通过存储区域网络 (SAN) 或任何形式的网络附连存储器 (NAS) 访问的任何类型的数据存储设备。

[0039] 在数据库系统 100 的替换实现中,可以在基于相同处理器的计算机系统上安装和

执行两个或更多砖。在图 3 中示出了这样的实现的框图。如图 3 所示,在基于单个处理器的计算机系统 302 上安装和执行多个砖  $102_1-102_m$ ,每一个砖包括对应的数据服务器实例和群集基础结构逻辑实例。安装在计算机系统 302 上的砖的数量——标示为  $m$ ,优选地小于数据库系统 100 中砖的总数——表示为  $n$ 。如在图 3 中进一步示出的,计算机系统 302 连接到通信基础结构 104,并连接到一个或多个数据存储设备 304。在一种实现中,数据存储设备 304 只能被计算机系统 302 访问。在这样的实现中,由数据库系统 100 的上下文内的数据库服务器实例  $112_1-112_m$  存储、检索、修改或以其他方式处理的任何数据库数据将被存储在附连到计算机系统 302 的数据存储设备 304 上。存储在数据存储设备 304 上的数据库数据在  $102_1-102_m$  之间不被共享。相反地,每一砖都具有其自己的对应的数据存储——在图 3 中标示为数据存储  $306_1-306_m$ 。例如,如果数据库数据被存储在数据存储设备 304 内的文件中,则每一文件对于砖  $102_1-102_m$  中的一个而言是独占的。作为另一个示例,如果数据库数据以原始存储格式来存储,则数据存储设备 304 内的物理盘对于砖  $102_1-102_m$  中的对应的一个而言是独占的。

[0040] 在图 1-3 中,通信基础结构 104 旨在表示能够将数据从一个计算机系统携带到另一计算机系统的任何通信基础结构。例如,在一种实现中,通信基础结构 104 包括使用吉比特以太网技术、InfiniBand® 技术等等来实现的高速局域网 (LAN)。然而,这些示例不旨在进行限制,而是可以使用其他通信基础结构。

[0041] 图 4 是数据存储设备 400 的框图,其可以表示如上文参考图 2 所讨论的数据存储设备 204 中的任一个,或如上文参考图 3 所讨论的数据存储设备 304 中的任一个。如图 4 所示,数据存储设备 400 包括多个存储位置  $402_1, 402_2, \dots, 402_i$ 。每一这样的存储位置可以包括,例如,可由与数据存储设备 400 所附连到的计算机系统相关联的文件系统标识和访问的卷。每一这样的存储位置也可以包括包含一个或多个卷的存储器的逻辑单元。每一这样的逻辑单元可以使用逻辑单元号 (LUN) 来标识。

[0042] 图 5 是更详细地示出了多个群集基础结构逻辑实例  $114_1-114_n$  的单个代表性实例 114 的框图。如图 5 所示,群集基础结构逻辑  $114_1-114_n$  的每一实例包括多个代理 502,并任选地包括一个或多个管理器 504。

[0043] 管理器 504 中的每一个被配置成控制允许多个数据库服务器实例  $112_1-112_n$  作为单个逻辑数据库系统一起操作所需的某些功能的性能,并允许与单个数据库相关联的数据被多个数据库服务器实例  $112_1-112_n$  同时存储、检索、修改或以其他方式处理。如图 6 所示,管理器 504 可包括配置管理器 602、数据虚拟化管理器 604、全局死锁管理器 606 和事务协调管理器 608 中的一个或多个。

[0044] 配置管理器 602 是关键群集管理器,并协调诸如对其他管理器和代理进的启动和关闭、对群集的重配置等等关键活动。

[0045] 数据虚拟化管理器 604 负责数据虚拟化。它关于应该将所有用户数据放置在哪里,以及应该将与这样的用户数据相关联的元数据放在哪里作出决定。出于达成可缩放性和避免瓶颈的目的,数据虚拟化管理器 604 还负责负载平衡。数据虚拟化管理器 604 实现用于以可缩放性换取数据的可用性和对齐的策略。

[0046] 在数据库系统 100 的一种实现中,如前所述的管理器类型中的每一个的实例只包括在群集基础结构逻辑  $114_1-114_n$  的  $n$  个实例的子集中。因此,例如,在其中  $n$  大于 2 的实

现中,数据虚拟化管理器 604 的实例可以被只包括在群集基础结构逻辑 114<sub>1</sub>-114<sub>n</sub> 的 2 个实例内。这用以节省资源,而且可以在万一当前正在执行的管理器发生故障的情况下有某一冗余度。每一管理器类型只有一个实例被允许在任何给定时间作出决策。每一管理器都被配置成通过向位于群集基础结构逻辑 114<sub>1</sub>-114<sub>n</sub> 的每一实例内的代理的对应的实例发送命令并从其接收信息来执行其指定功能。如图 7 所示,这些代理 504 包括配置管理器代理 702、数据虚拟化管理器代理 704、全局死锁管理器代理 706 和事务协调管理器代理 708。

[0047] 数据库系统 100 部分地通过提供在多个不同的计算机系统上执行的多个数据库服务器实例 112<sub>1</sub>-112<sub>n</sub> 来实现高可用性,每一个数据库服务器实例可以被用来访问单个逻辑数据库。如果数据库服务器实例或它在其上执行的计算机系统发生故障,则可以使用在不同的计算机系统上执行的一个或多个其他数据库服务器实例来获取数据库服务。

[0048] 数据库系统 100 通过以下操作来实现增强的性能:跨与在其上执行砖 102<sub>1</sub>-102<sub>n</sub> 的不同计算机系统相关联的多个数据存储设备存储来自数据库的数据,以使得与处理这样的数据相关联的工作负载可以跨多个计算机系统地分布。数据库系统 100 通过跨这样的数据存储设备存储相同数据库数据的副本进一步实现高可用性,以使得如果一个计算机系统和 / 或与其相关联的数据存储设备发生故障,则可以通过不同的计算机系统和相关联的数据存储设备来访问相同数据的替换性副本。现在将参考图 8-10 例示这些概念。

[0049] 具体而言,图 8 描绘了数据库中包括诸如示例性行 812 之类的一系列行的表 802。每一数据库服务器实例 112<sub>1</sub>-112<sub>n</sub> 被配置成向用户提供用于创建这样的表的能力,以及此外用于分割这样的表以产生被叫做分区的行的组。例如,如图 8 中所进一步示出的,表 802 可以被分割成第一分区 804 和第二分区 806。

[0050] 数据虚拟化管理器 604 被配置成将每一分区进一步分割成被叫做“段”的较小的行的组。例如,如图 9 所示,第一分区 804 可以被分割成第一段 902,第二段 904 和第三段 906。段是逻辑实体。段的物理表示被称为克隆。

[0051] 数据虚拟化管理器 604 进一步被配置成跨与不同的计算机系统相关联的数据存储设备分布克隆,以改善性能和提供高可用性。数据虚拟化管理器 604 可以基于冗余因子来确定要创建和跨数据存储设备分布的克隆的数量。可以由系统管理员或用户取决于实现来设置冗余因子。

[0052] 例如,图 10 是被表示为数据库系统 1000 的数据库系统 100 的一个实现的框图,其中,克隆跨与不同的计算机系统相关联的数据存储设备分布。如图 10 所示,执行砖 1014 的计算机系统 1010 连接到数据存储设备 1012,执行砖 1024 的计算机系统 1020 连接到数据存储设备 1022,而执行砖 1034 的计算机系统 1030 连接到数据存储设备 1032。计算机系统经由通信基础结构 1004 连接。假设图 9 的第一段 902 在物理上被表示为克隆 1002<sub>1</sub>、1002<sub>2</sub> 和 1002<sub>3</sub>,图 9 的第二段 904 在物理上被表示为克隆 1004<sub>1</sub>、1004<sub>2</sub> 和 1004<sub>3</sub>,而图 9 的第三段 906 在物理上被表示为克隆 1006<sub>1</sub>、1006<sub>2</sub> 和 1006<sub>3</sub>。

[0053] 如图 10 所示,数据虚拟化管理器 604 将与每一段相关联的一个克隆分别分布到数据存储设备 1012、1022 和 1032 中的每一个。例如,克隆 1002<sub>1</sub> 被存储在数据存储设备 1012 内,克隆 1002<sub>2</sub> 被存储在数据存储设备 1022 内,而克隆 1002<sub>3</sub> 被存储在数据存储设备 1032 内。结果,与对构成第一分区 804 的所有三个段 902、904 和 906 操作的任何过程相关联的工作负载可以容易地跨计算机系统 1010、1020、1030 分布,因为每一计算机系统都具有对

必要的数据的本地访问,以便执行过程。此外,如果计算机系统 1010、1020、1030 中的任一个或其相关联的数据存储设备万一发生故障,则由段 902、904 和 906 逻辑表示的数据仍可经由其他计算机系统和相关联数据存储设备中的任一个来访问。

[0054] 数据库系统 1000 的体系结构可以被称为“不共享任何东西”体系结构,因为系统 1000 内的每一计算机系统不与其他计算机系统中的任一个计算机系统共享任何共用资源来访问和处理必需的数据库数据。体系结构有利地通过添加新计算机系统和数据存储设备来方便地横向扩展。

[0055] B. 存储层

[0056] 某些常规数据库服务器要求用户指定与特定数据库相关联的数据将被存储在哪里的物理位置。存储指定可包括,例如,一个或多个数据库文件。作为数据库创建过程的一部分,用户可能需要指定物理存储位置。

[0057] 如前面的章节所描述的将这样的一个方案延伸到数据库系统 100 会造成许多问题。例如,如果数据库的创建者需要指定与数据库相关联的数据将如何存储在与在其上执行砖 102<sub>1</sub>-102<sub>n</sub> 的计算机系统相关联的各种数据存储设备中,则数据库系统 100 的单个系统镜像宗旨将被违犯。

[0058] 此外,如果数据库系统 100 被纵向扩展以包括更大数量的计算机系统和更大数量的相关联数据存储设备,则与跨所有数据存储设备指定存储位置相关联的复杂性同等地增大。

[0059] 另外,如上文所示的,与数据库系统 100 相关联的目标是高可用性。这在数据库系统 100 中部分地通过跨与多个不同的计算机系统相关联的多个不同的数据存储设备协调地创建和存储相同的数据库数据的多个表示来实现。这种创建和存储方案允许无缝地处理诸如砖的故障之类的问题。允许用户指定与数据库相关联的数据将被存储在哪里的准确物理位置可以阻碍或禁用这样的自动化创建和存储功能。

[0060] 更进一步,在其中用户需要指定与数据库相关联的数据将被存储在哪里的物理位置的数据库系统中,用户可能需要处理当与单个数据库相关联的多个文件被存储在不同的物理位置时出现的文件名激增 (proliferation) 的问题。

[0061] 本发明的一个实施例通过提供全系统范围的逻辑存储容器 (被称为存储层) 来解决前述问题中的每一个问题。每一存储层都逻辑地表示一个或多个存储位置。由存储层逻辑表示的存储位置可以存在在多个不同的数据存储设备内,其中多个不同的数据存储设备中的每一个只能够被一组互连的计算机系统中的对应的计算机系统来访问。使用存储层有利地使得诸如数据库系统 100 之类的系统能向作为数据库系统 100 的一部分的每一砖上的用户呈现单个系统映像。

[0062] 通过提供存储层,本发明的一个实施例为存储器提供可以由用户直接地处理的单个系统抽象。因此,用户不必顾虑细粒度细节和与跨大量的数据存储设备存储数据相关联的复杂性。这样的单个系统抽象给用户提供在处理数据库系统范围内的存储要求时使用、经营、和管理的简便性。此外,与存储层合作时涉及的复杂性有利地保持恒定,而不管数据库系统的大小如何。

[0063] 使用存储层还允许诸如数据虚拟化管理器 604 之类的软件实体负责跨多个不同的数据存储设备创建和存储数据库数据。结果,用户不必顾虑指定与数据库相关联的数据

将被存储在哪里的准确物理位置。用户也不用担心关于文件名扩散问题,因为,在一个实施例中,文件被系统软件实体自动地命名。

[0064] 1. 数据库文件和文件组

[0065] 为提供对存储层的属性和使用的更好理解,现在将描述可以与根据本发明的一实施例的存储层相关联的各种类别的数据库数据。此描述与数据库系统 100 的一个实施例特别相关,其中,数据库服务器 112<sub>1</sub>-112<sub>n</sub> 的每一实例都包括由美国华盛顿州雷德蒙市的微软公司发布的Microsoft® SQL Server®的一个版本。然而,本发明不仅限于这样的实施例。

[0066] 数据库系统 100 中的数据库可以具有三种类型的文件:主数据文件、辅助数据文件和日志文件。主数据文件是数据库的起始点,并指向数据库中的其他文件。每一数据库具有一个主文件。对于主数据文件的推荐的文件名扩展是 .mdf。

[0067] 辅助数据文件构成除主数据文件以外的与数据库相关联的所有数据文件。某些数据库可以不具有任何辅助数据文件,而其他数据库具有若干个辅助数据文件。对于辅助数据文件的推荐的文件名扩展是 .ndf。

[0068] 日志文件保留被用来恢复数据库的所有日志信息。对于每一个数据库,必须有至少一个日志文件,虽然可以有一个以上的日志文件。对于日志文件的推荐的文件名扩展是 .ldf。

[0069] 在数据库系统 100 中,可以将数据库对象和文件编组在文件组中,以便用于分配和管理的目的。有两种类型的文件组:主文件组和用户定义文件组。与数据库相关联的主文件组包含主数据文件和未被专门指派给另一文件组的任何其他文件。系统表(下面将讨论)的所有页面被分配在主文件组中。用户定义文件组是通过在 CREATE DATABASE(创建数据库)或 ALTER DATABASE(更改数据库)语句中使用 FILEGROUP(文件组)关键字来指定的任何文件组。

[0070] 日志文件决不会是文件组的一部分。日志空间被与数据空间分开地管理。

[0071] 没有文件可以是一个以上的文件组的成员。表、索引,以及大型对象数据可以与指定的文件组相关联。在此情况下,所有页面都将被分配在该文件组中,或者,表和索引都可以被分区。经分区的表和索引的数据被划分成若干个单元,其中每一个单元可以被置于数据库中的单独文件组中。

[0072] 每一数据库中的一个文件组被指定为默认文件组。当创建表或索引而不指定文件组时,假设所有页面将从默认文件组分配。一次只有一个文件组可以是默认文件组。db\_owner(数据库\_所有者)固定数据库角色的成员可以将默认文件组从一个文件组切换到另一文件组。如果没有指定默认文件组,则主文件组是默认文件组。

[0073] 与数据库系统 100 相关联的系统元数据可以被存储在数个系统数据库中,其中每一个系统数据库具有数个前述文件类型。例如,系统元数据可包括 master 数据库和模型数据库,其中每一个都包括数据和日志文件。系统表中有三种元数据:配置管理器、事务协调管理器和数据虚拟化管理器的逻辑元数据、物理元数据和持久状态 / 元数据。

[0074] 逻辑元数据是被复制或在物理上被保存到与数据库系统 100 中的每一个砖相关联的数据存储设备的数据。一个叫做“元数据管理器”的软件实体被配置成执行此功能。

[0075] 物理元数据描述存储在只能被特定砖在其上执行的计算机系统访问的数据存储设备上的元数据。没有被复制的副本,且系统表被建模为在每一数据库段上具有单独的数

据段。如此,这些表的内容是相对于每一砖本地存储的所有物理元数据的并。

[0076] 根据预定义的算法将配置管理器 / 事务协调管理器 / 数据虚拟化管理器元数据复制到与某些砖相关联的数据存储设备。从元数据管理器观点来看,此元数据被视为“物理元数据”。

## [0077] 2. 存储层的属性

[0078] 在下表 1 中提供了根据本发明的一个实施例的对存储层的每一实例通用的属性的描述。将描述的一些属性与数据库系统 100 的一个实施例特别相关,其中,数据库服务器 112<sub>1</sub>-112<sub>n</sub> 的每一实例包括由华盛顿州雷德蒙市的微软公司发布的 Microsoft® SQL Server® 的一个版本,虽然存储层的用途不仅限于这样的实施例。

[0079]

属性	值	备注
storage_tier_id (存储层_id)	[1,k],其中, k 是 4 字节整数值。	系统生成的值。不可改变的属性。在给定数据库系统上是唯一的。
名称	遵循对象命名约定的任何名称。	由系统为默认实例提供的。用户为附加实例提供的名称。可使用 ALTER STORAGE TIER (更改存储层) 命令来更新。在给定数据库系统上是唯一的。
类型	{ <i>system_data</i> (系统_数据), <i>system_log</i> (系统_日志), <i>temp_data</i> (临时_数据), <i>temp_log</i> (临时_日志), <i>data</i> (数据), <i>log</i> (日志) }	在实例创建期间设置。随后无法被修改。
is_default(为_默认)	布尔	在数据库系统中总有一个并且正好一个给定存储层类型的默认实例。
storage_pool (存储池)	存储器规范集合	可使用 ALTER STORAGE TIER 命令来更新。

[0080] 表 1 :存储层的描述

[0081] 如表 1 所示,存储层实例的属性被标记为 storage\_tier\_id、名称、类型、is\_default 和 storage\_pool。storage\_tier\_id 属性包括由数据库系统 100 内的软件实体生成的唯一地标识数据库系统 100 内的每一砖的单个存储层实例的不可改变的值。

[0082] 名称属性包括与数据库系统 100 内的每一砖的存储层实例唯一地相关联的名称。名称可能需要遵循与数据库服务器实例 112<sub>1</sub>-112<sub>n</sub> 相关联的对象命名约定，诸如结构化查询语言 (SQL) 对象命名约定。在数据库系统 100 的一种实现中，系统为每一种类型的存储层提供默认存储层实例。在这样的实现中，由数据库系统 100 提供与默认存储层实例相关联的名称，而相比之下，所有用户创建的存储层实例都由用户命名。在一个实施例中，用于命名存储层的命名空间是平坦非分层命名空间。如表 1 所指出的，可以使用 ALTER STORAGE TIER 命令来更新与存储层实例相关联的名称，如此处比较详细地描述的。

[0083] 存储层的每一实例具有在存储层实例的创建期间设置的类型属性。一旦设置，就无法修改指派给存储层实例的类型。存储层类型包括，但不仅限于，system\_data、system\_log、temp\_data、temp\_log、data 和 log。下面将更详细地描述这些存储层类型。

[0084] 属性 is\_default 指定存储层实例是否是存储层的默认实例。在一个实施例中，给定存储层类型只有一个默认实例。

[0085] 属性 storage\_pool 标识与存储层实例相关联的一个或多个存储规范。在下表 2 中描述了根据本发明的一个实施例的存储规范的示例。如表 2 所示，与存储规范实例相关联的属性包括 storage\_tier\_id(存储层\_id)、storage\_spec\_id(存储空间\_id)、brick\_id(砖\_id) 和路径。

[0086]

属性	值	备注
storage_tier_id	[1,m], 其中，m 是 4 字节整数值。	不可改变的属性。
storage_spec_id	[1, k], 其中，k 是 4 字节整数值	(storage_tier_id, storage_spec_id) 是合成键，并且在数据库系统上是唯一的。
storage_spec_name (存储空间_名称)		应该遵循用于命名数据库服务器中的标识符的规则。在给定存储层上是唯一的。
brick_id	[1,n], 4 字节整型	
路径	<到目录的路径>	路径应该始终以尾随反斜线结束。

[0087] 表 2 :存储规范的描述

[0088] 属性 storage\_tier\_id 是唯一地标识存储规范与其相关联的存储层实例的不可改变的值。

[0089] 属性 storage\_spec\_id 是与由 storage\_tier\_id 标识的存储层实例有关的唯一地标识存储规范实例的值。如表 2 所指出的，storage\_tier\_id 和 storage\_spec\_id 的组合定义唯一地标识数据库系统 100 内的所有砖的存储规范的组合键。

[0090] 属性 storage\_spec\_name 包括与存储规范相关联的名称。storage\_spec\_name 在任何给定存储层实例上必须是唯一的，并可能需要遵循用于命名与数据库服务器实例 112<sub>1</sub>-112<sub>n</sub> 相关联的标识符的某些规则。

[0091] 属性 brick\_id 是数据库系统 100 内砖 102<sub>1</sub>-102<sub>n</sub> 中存储规范与之相关联的一个的唯一标识符。

[0092] 属性“路径”描述了至与由 brick\_id 标识的砖在其上执行的计算机系统相关联的数据存储设备内的存储位置的路径。如上文参考图 4 所讨论的，存储位置可以包括，例如，可由与计算机系统相关联的文件系统识别和访问的卷。也如上文参考图 4 所讨论的，存储位置可以包括包含一个或多个卷的存储的逻辑单元，其中逻辑单元可以通过逻辑单元号 (LUN) 来标识。

### [0093] 3. 存储层类型

[0094] 如上文参考表 1 所讨论的，每一存储层实例具有类型属性。与存储层相关联的类型确定该存储层的属性的数量，包括，但不仅限于，可以为该类型创建的存储层实例的数量，是否可以由用户创建或丢弃存储层的实例，以及可以与存储层的实例相关联的数据库文件的类型。

[0095] 下表 3 标识了根据本发明的一个实施例的存储层实例的不同类型。下面将描述与这些不同的存储层类型中的每一个相关联的属性。

[0096]

类型	系统提供的实例的名称	数据库系统中的实例的数量
<i>system_data</i> (系统_数据)	StSystemData (St 系统数据)	1 (仅系统提供)
<i>system_log</i> (系统_日志)	StSystemLog (St 系统日志)	1 (仅系统提供)
<i>temp_data</i> (临时_数据)	StTempData (St 临时数据)	1 (仅系统提供)
<i>temp_log</i> (临时_日志)	StTempLog (St 临时日志)	1 (仅系统提供)
<i>Data</i> (数据)	StData (St 数据)	用户可以创建任何数量
<i>Log</i> (日志)	StLog (St 日志)	用户可以创建任何数量

[0097] 表 3 :存储层类型

[0098] system data 和 system log 存储层类型的属性。在任何给定时间可以有一个且只有一个类型 system\_data 和 system\_log 的存储层的实例。这些实例分别带有名称 StSystemData 和 StSystemLog，并由数据库系统 100 提供。这些存储层实例控制对与数据库系统 100 相关联的系统元数据的存储器分配。具体而言，存储层实例 StSystemData 控制对于构成系统元数据的数据库（例如，主（master）数据库和模型数据库）的数据文件的存

储分配,而存储层实例 StSystemLog 控制对于构成系统元数据的数据库的日志文件的存储分配。在本发明的一个实施例中,必须在与数据库系统 100 中的每一个砖相关联的一个或多个数据存储设备上预设系统元数据的存储。

[0099] 用户不允许丢弃类型 system\_data 和 system\_log 的存储层的系统提供的实例。用户也不能创建类型 system\_data 和 system\_log 的存储层实例。用户可以预设更多存储或更改与存储层类型 system\_data 和 system\_log 的系统提供的实例相关联的预设存储。

[0100] 对于类型 system\_data 和 system\_log 的每一个存储层实例, is\_default 属性的值为真,并且无法被更改。

[0101] temp\_data 和 temp\_log 存储层类型的属性。在任何给定时间可以有一个且只有一个类型 temp\_data 和 temp\_log 的存储层的实例。这些实例分别带有名称 StSystemData 和 StSystemLog,并由数据库系统 100 提供。在一个实施例中, tempdb(临时数据库)描述了每一数据库服务器实例 1121-112n 的恰当操作所需的且在数据库系统 100 的一个实施例中在全局级别下提供的(即,供所有砖使用)临时数据库。存储层实例 StTempData 控制对于 tempdb 的主文件组的存储分配,而存储层实例 StTempLog 控制对于 tempdb 的日志文件的存储分配。在本发明的一个实施例中,必须在与数据库系统 100 中的每一个砖相关联的一个或多个数据存储设备上预设用于 tempdb 数据和日志文件的存储。

[0102] 用户不允许丢弃类型 temp\_data 和 temp\_log 的存储层的系统提供的实例。用户也不能创建类型 temp\_data 和 temp\_log 的存储层实例。用户可以预设更多存储或更改与存储层类型 temp\_data 和 temp\_log 的系统提供的实例相关联的预设的存储。

[0103] 对于类型 temp\_data 和 temp\_log 的每一个存储层实例, is\_default 属性的值为真,并且无法被改变。

[0104] data 和 log 存储层类型的属性。类型 data 的存储层实例控制对与用户创建数据库相关联的数据文件的存储分配,而类型 log 的存储层实例控制对与用户创建数据库相关联的日志文件的存储分配。用户可以只创建 data 和 log 存储层类型的实例。可以创建任意数量的实例。在一个实施例中,可以在与数据库系统 100 中的任何砖相关联的任何数据存储设备(诸)上预设数据和日志文件。在又一实施例中,必须在于其上为给定用户创建数据库的数据文件预设存储的相同砖上预设该用户创建数据库的日志文件的存储。

[0105] 如果没有数据库当前链接到存储层实例,则类型 data 或类型 log 的存储层的实例可以被用户丢弃。在一个实施例中,数据库系统 100 维护标识当前链接到存储层实例的数据库的数量的与每一存储层实例相关联的属性,表示为 RefCount(参考计数)。如此,只有当与实例相关联的 RefCount 等于零时,才可以丢弃类型 data 或类型 log 的存储层实例。

[0106] 为这些存储层类型中的每一种类型提供了系统提供的默认实例。对于类型 data 的系统提供的默认实例被命名为 StData,对于类型 log 的系统提供的默认实例被命名为 StLog。数据库系统 100 最初将这些默认实例的 is\_default 值设置为真。当这些类型中的任一个类型的新实例被选择为默认时,数据库系统 100 自动地在前面的默认实例上将 is\_default 的值标记为假。如此,在任何时间,在数据库系统 100 中可以有一个且只有一个每一存储层类型的默认实例。

[0107] 4. 存储层实例的创建、更改和丢弃

[0108] 现在将描述可以创建、更改或丢弃存储层实例的方式。这些功能可以由数据库系

统 100 的任何用户来执行,虽然可以预料这样的功能通常将由数据库管理员 (DBA)、存储管理员,或负责数据库系统 100 的管理的其他授权人员来执行。

[0109] 图 11 是描绘了在执行涉及存储层实例的创建、更改或丢弃的功能时可能涉及的实体的框图。如图 11 所示,这些实体包括砖 102<sub>1</sub>,该砖 102<sub>1</sub> 包括如上文参考图 1 所讨论的数据库服务器实例 112<sub>1</sub> 和群集基础结构逻辑实例 114<sub>1</sub>,虽然可以使用数据库系统 100 中的任何其他砖。客户机 1102 可通信地连接到数据库服务器实例 112<sub>1</sub>。这样的连接可以通过通信基础结构 104 或经由某种其他通信基础结构来建立。群集基础结构逻辑实例 114<sub>1</sub> 提供对逻辑系统元数据 1104 的访问,该逻辑系统元数据 1104 如上文所讨论的,被复制或在物理上被保存到与数据库系统 100 中的每一砖相关联的数据存储设备。

[0110] 如在图 11 中进一步示出的,数据库服务器实例 112<sub>1</sub> 包括命令处理器 1112 和元数据管理器 1114。命令处理器 1112 是被配置成接收和处理由客户机 1102 的用户提交的命令的软件逻辑,其中这样的命令可包括用于创建、更改或丢弃存储层的命令。客户机 1102 提供用户可以用来提交这样的命令的用户界面。在一个实施例中,命令包括事务-SQL(T-SQL)命令,虽然本发明没有这样的限制。

[0111] 元数据管理器 1114 包括部分地被配置为响应于由命令处理器 1112 对某些命令的处理,创建、修改或删除与存储层相关联的元数据的软件逻辑。与存储层相关联的元数据被存储为逻辑系统元数据 1104 的一部分。由于逻辑系统元数据 1104 在物理上被保存到与数据库系统 100 中的每一砖相关联的数据存储设备,因此由元数据管理器 1114 对这样的元数据的创建、修改或删除在群集基础结构逻辑实例 114<sub>1</sub> 的协助下执行。

[0112] 图 12 描绘了创建存储层可以所采用的示例方法的流程图 1200。流程图 1200 的步骤是只作为示例来描述的,且不旨在限制本发明。此外,虽然可以参考各种逻辑和 / 或物理实体和在本文中的别处描述的系统来描述流程图 1200 的步骤,但是,精通相关技术的人员将轻松地理解,方法不必一定使用这样的实体和系统来实现。

[0113] 如图 12 所示,流程图 1200 的方法从步骤 1202 开始,在该步骤中,命令处理器 1112 接收存储层实例的标识符。存储层实例的标识符可以包括例如将被指定给存储层实例的名称。存储层实例的标识符可以作为由客户机 1102 的用户向数据库服务器实例 112<sub>1</sub> 提交的命令——诸如 T-SQL 命令——的部分来接收。

[0114] 在步骤 1204 中,命令处理器 1112 接收一个或多个存储位置的标识符。存储位置可以包括,例如,多个数据存储设备中的每一个内的一个或多个存储位置,其中多个数据存储设备中的每一个可分别被数据库系统 100 内的不同的砖来访问。如在别处所指出的,在某些实施例中,存储位置可以包括卷或标识一个或多个卷的 LUN(逻辑单元号)。存储位置的标识符可以包括,例如,到目录的路径。一个或多个存储位置的标识符可以作为由客户机 1102 的用户向数据库服务器实例 112<sub>1</sub> 提交的命令的——诸如 T-SQL 命令——的部分来提供。命令可以是用来在步骤 1202 中提供存储层实例的标识符的相同命令。

[0115] 在步骤 1206 中,响应于在步骤 1202 中接收到存储层实例的标识符和在步骤 1204 中接收到一个或多个存储位置的标识符,命令处理器 1112 将在步骤 1202 中标识的存储层实例与在步骤 1204 中标识的存储位置相关联,以使得存储层实例逻辑地表示存储位置。命令处理器 1112 可以,例如,响应于接收到包括存储层实例的标识符和存储位置的标识符的命令——诸如 T-SQL 命令,执行此步骤。一旦进行了前述的关联,由元数据管理器 1114 通

过使用群集基础结构逻辑实例 114<sub>1</sub> 将表示关联的元数据存储为逻辑系统元数据 1104 的部分。

[0116] 一旦根据前述的方法创建了存储层实例, 它就可以被数据虚拟化管理器 604 用来自动地将来自系统或与实例相关联的用户创建数据库的数据存储在由实例所标识的存储位置。在一个实施例中, 系统数据库文件和存储层实例之间的关系是由数据库系统 100 建立的, 而用户创建数据库文件和存储层实例之间的关系可以由数据库系统 100 建立或者由用户作为数据库创建过程的一部分来建立。

[0117] 下面提供了用于创建存储层实例的示例命令句法:

[0118]

```
CREATE STORAGE TIER storage_tier_name OF TYPE type_name
    [ ADD <storage_spec> [, ...]
    ;]

<storage_spec>:= (name = storage_spec_name, brick_id = value, path
= path_to_directory)
```

[0119] 在前述的命令中, storage\_tier\_name(存储\_层\_名称) 是标识要创建的存储层实例的名称, type\_name(类型\_名称) 标识要创建的存储层实例的类型, 而 <storage\_spec>(存储\_空间) 标识要由存储层实例来逻辑表示的存储位置。在一个实施例中, type\_name(类型\_名称) 可以是 data 或 log 中的一个, 其中这样的类型对应于上文参考表 3 所描述的 data 和 log 类型。在又一实施例中, <storage\_spec> 对应于如上文参考表 2 所描述的存储规范。

[0120] 下面是用于创建类型 log 的新的用户定义存储层实例的前述命令句法的使用的示例:

[0121]

```
CREATE STORAGE TIER StLog2 of TYPE LOG
    ADD (NAME = WDRIVE, BRICK_ID = 100, PATH = 's:\',
    go
```

[0122] 图 13 中描绘了藉由其可以更改现有的存储层实例以便将一个或多个新存储位置与存储层实例相关联的示例方法的流程图 1300。流程图 1300 的步骤是只作为示例来描述的, 而不旨在限制本发明。此外, 虽然可以参考各种逻辑和 / 或物理实体和在本文中的别处所描述的系统来描述流程图 1300 的步骤, 但是, 精通相关技术的人员将轻松地理解, 方法不必一定使用这样的实体和系统来实现。

[0123] 如图 13 所示, 流程图 1300 的方法从步骤 1302 开始, 在该步骤中, 命令处理器 1112 接收存储层实例的标识符。存储层实例的标识符可以包括例如已被指派给存储层实例的名称。存储层实例的标识符可以作为由客户机 1102 的用户向数据库服务器实例 1121 提交的命令——诸如 T-SQL 命令——的部分来接收。

[0124] 在步骤 1304 中,命令处理器 1112 接收未由存储层实例逻辑表示的至少一个存储位置的标识符。至少一个存储位置可以包括,例如,数据存储设备内可被数据库系统 100 内的特定砖访问的存储位置。至少一个存储位置的标识符可以包括,例如,到目录的路径。至少一个存储位置的标识符可以作为由客户机 1102 的用户向数据库服务器实例 112<sub>1</sub> 提交的命令——诸如 T-SQL 命令——的部分来提供。命令可以是用来在步骤 1302 中提供存储层实例的标识符的相同命令。

[0125] 在步骤 1306 中,响应于在步骤 1302 中接收到存储层实例的标识符和在步骤 1304 中接收到至少一个存储位置的标识符,命令处理器 1112 将在步骤 1304 中标识的至少一个存储位置与在步骤 1302 中标识的存储层实例相关联,以使得存储层实例逻辑地表示至少一个存储位置。命令处理器 1112 可以,例如,响应于接收到包括存储层实例的标识符和至少一个存储位置的标识符的命令——诸如 T-SQL 命令,执行此步骤。一旦发生了前述的关联,元数据管理器 1114 就对与存储层实例相关联的并且存储在逻辑系统元数据 1104 中的元数据作出相对应的修改,其中这样的修改是使用群集基础结构逻辑实例 114<sub>1</sub> 来作出的。

[0126] 一旦根据前述方法改变了存储层实例,数据虚拟化管理器 604 就可以将来自己被指派给存储层实例的数据库文件的数据自动地存储在相关联的存储位置。

[0127] 图 14 中描绘了藉由其可以更改现有存储层实例以便将一个或多个存储位置与存储层实例解除关联的示例方法的流程图 1400。流程图 1400 的步骤是只作为示例来描述的,不旨在限制本发明。此外,虽然可以参考各种逻辑和 / 或物理实体和在本文中的别处所描述的系统来描述流程图 1400 的步骤,但是,精通相关技术的人员将轻松地理解,方法不必一定使用这样的实体和系统来实现。

[0128] 如图 14 所示,流程图 1400 的方法从步骤 1402 开始,在该步骤中,命令处理器 1112 接收存储层实例的标识符。存储层实例的标识符可以包括例如已被指派给存储层实例的名称。存储层实例的标识符可以作为由客户机 1102 的用户向数据库服务器实例 112<sub>1</sub> 提交的命令——诸如 T-SQL 命令——的部分来接收。

[0129] 在步骤 1404 中,命令处理器 1112 接收由存储层实例逻辑表示的至少一个存储位置的标识符。至少一个存储位置可以包括,例如,数据存储设备内可被数据库系统 100 内的特定砖访问的存储位置。至少一个存储位置的标识符可以包括例如到目录的路径。至少一个存储位置的标识符可以作为由客户机 1102 的用户向数据库服务器实例 112<sub>1</sub> 提交的命令——诸如 T-SQL 命令——的部分来提供。命令可以是用来在步骤 1402 中提供存储层实例的标识符的相同命令。

[0130] 在步骤 1406 中,响应于在步骤 1402 中接收到存储层实例的标识符和在步骤 1404 中接收到至少一个存储位置的标识符,命令处理器 1112 将在步骤 1404 中标识的至少一个存储位置 1404 与在步骤 1402 中标识的存储层实例解除关联,以使得存储层实例不再逻辑地表示至少一个存储位置。命令处理器 1112 可以,例如,响应于接收到包括存储层实例的标识符和至少一个存储位置的标识符的命令——诸如 T-SQL 命令,执行此步骤。一旦发生了前述的解除关联,元数据管理器 1114 就对与存储层实例相关联的并且存储在逻辑系统元数据 1104 中的元数据作出相对应的修改,其中这样的修改是使用群集基础结构逻辑实例 114<sub>1</sub> 来作出的。

[0131] 一旦根据前述的方法更改了存储层实例,数据虚拟化管理器 604 就可以从解除关

联的存储位置自动地移除已被指派给存储层实例的数据库文件的数据。

[0132] 下面提供了用于更改存储层实例的示例命令句法：

[0133]

```
ALTER STORAGE TIER storage_tier_name
  [ ADD <storage_spec> [, ...n] ]
  [ REMOVE STORAGE_SPEC = storage_spec_name [, ...n] ]
  [ MODIFY Name = new_storage_tier_name]
[;]
```

[0134] 在前述的命令中, `storage_tier_name`(存储\_层\_名称)是标识要更改的存储层实例的名称。ADD(添加)、REMOVE STORAGE\_SPEC(移除存储\_空间),以及 MODIFY(修改)子命令可以各自被包括在 ALTER STORAGE TIER(更改存储层)命令内,以分别向存储层添加存储位置,从存储层删除存储位置,或修改存储层名称。

[0135] 下面是用于向类型 `data` 的默认存储层实例预设某一存储的前述命令句法的使用的一个示例：

[0136]

```
ALTER STORAGE TIER StData
  ADD (NAME = CDRIVE, BRICK_ID = 100, PATH = 'c:\',
        ADD (NAME = XDRIVE, BRICK_ID = 100, PATH = 'x:\')
go
```

[0137] 下面是用于向类型 `log` 的默认存储层实例预设某一存储的前述命令句法的使用的另一示例：

[0138]

```
ALTER STORAGE TIER StLog
  ADD (NAME = SDRIVE, BRICK_ID = 100, PATH = 's:\',
        ADD (NAME = TDRIVE, BRICK_ID = 100, PATH = 't:\')
go
```

[0139] 图 15 描绘了藉由其可以丢弃现有存储层实例的示例方法的流程图。流程图 1500 的步骤是只作为示例来描述的,且不旨在限制本发明。此外,虽然可以参考各种逻辑和 / 或物理实体和在本文中的别处所描述的系统来描述流程图 1500 的步骤,但是,精通相关技术的人员将轻松地理解,方法不必一定使用这样的实体和系统来实现。

[0140] 如图 15 所示,流程图 1500 的方法从步骤 1502 开始,在该步骤中,命令处理器 1112 接收存储层实例的标识符。存储层实例的标识符可以包括例如已被指派给存储层实例的名称。存储层实例的标识符可以作为由客户机 1102 的用户向数据库服务器实例 112 提交的命令——诸如 T-SQL 命令——的部分来接收。

[0141] 在步骤 1504 中,命令处理器 1112 确定是否有当前与在步骤 1502 中标识的存储层

实例相关联的任何数据库。在一个实施例中，数据库系统 100 维护与每一存储层实例相关联的、标识当前链接到存储层实例的数据库的数量的属性，标示为 RefCount。如此，命令处理器 1112 可以通过分析 RefCount 的值来确定是否有当前与在步骤 1502 中标识的存储层实例相关联的任何数据库。如果 RefCount 大于 0，那么，一个或多个数据库当前与存储层实例相关联。如果 RefCount 等于 0，那么，没有数据库当前与存储层实例相关联。

[0142] 在步骤 1506 中，响应于在步骤 1502 中接收到存储层实例的标识符并确定没有数据库当前与标识的存储层实例相关联，命令处理器 1112 丢弃所标识的存储层实例。命令处理器 1112 可以，例如，响应于接收到包括存储层实例的标识符的命令——诸如 T-SQL 命令，执行此步骤。一旦丢弃了存储层实例，元数据管理器 1114 就从逻辑系统元数据 1104 中删除与存储层实例相关联的元数据，其中这样的删除是使用群集基础结构逻辑实例 114<sub>1</sub> 来执行的。

[0143] 下面提供了用于丢弃存储层实例的示例命令句法：

[0144]

```
DROP STORAGE TIER storage_tier_name
```

```
[;]
```

[0145] 在前述命令中，storage\_tier\_name 是标识要丢弃的存储层实例的名称。

[0146] 5. 将数据库指派给存储层实例，并据此进行存储

[0147] 现在将描述数据库可被指派给存储层实例并据此进行存储的方式。特定数据库和特定存储层实例之间的关联可以由数据库系统 100 自动地预设，或者可以由用户作为数据库创建过程的部分来创建。跨由存储层实例逻辑表示的一个或多个存储位置存储来自数据库的数据是由诸如以上文参考图 6 所描述的数据虚拟化管理器 604 之类的数据虚拟化管理器自动地处理的过程。

[0148] 图 16 是描绘了在执行与将数据库指派给存储层实例以及据此存储来自数据库的数据有关的功能时可能涉及的实体的框图。如图 16 所示，这些实体包括砖 102<sub>1</sub>，该砖 102<sub>1</sub> 包括如上文参考图 1 所讨论的数据库服务器实例 112<sub>1</sub> 和群集基础结构逻辑实例 114<sub>1</sub>，虽然可以使用数据库系统 100 中的任何其他砖。客户机 1602 通信地连接到数据库服务器实例 112<sub>1</sub>。这样的连接可以通过通信基础结构 104 或经由某种其他通信基础结构来建立。

[0149] 如在图 11 中进一步示出的，数据库服务器实例 112<sub>1</sub> 包括命令处理器 1112。命令处理器 1112 是被配置成接收和处理由客户机 1602 的用户提交的命令的软件逻辑，其中这样的命令可包括与数据库的创建有关的命令。客户机 1602 提供用户可以用来提交这样的命令的用户界面。在一个实施例中，命令包括 T-SQL 命令，虽然本发明没有这样的限制。

[0150] 群集基础结构逻辑实例 114<sub>1</sub> 包括数据虚拟化管理器 1612，该数据虚拟化管理器 1612 被配置成将来自所创建的数据库的数据存储在由与数据库相关联的存储层实例逻辑表示的多个存储位置 1604 中的任一个。每一个存储位置可以位于可以被数据库系统 100 内的不同的计算机系统访问的不同的数据存储设备内。

[0151] 在一替换实施例中，数据虚拟化管理器 1612 被包括在数据库系统 100 中除群集基础结构逻辑实例 114<sub>1</sub> 以外的群集基础结构逻辑的实例内，且群集基础结构逻辑实例 114<sub>1</sub> 包括数据虚拟化管理器代理，该代理被配置成与之进行通信，以使数据虚拟化管理器 1612 执

行前述功能。

[0152] 在图 17 中描绘了可以将数据库与存储层实例相关联以及据此存储来自数据库的数据的示例方法的流程图 1700。流程图 1700 的步骤是只作为示例来描述的,而不旨在限制本发明。此外,虽然可以参考各种逻辑和 / 或物理实体和在本文中的别处所描述的系统来描述流程图 1700 的步骤,但是,精通相关技术的人员将轻松地理解,方法不必一定使用这样的实体和系统来实现。

[0153] 如图 17 所示,流程图 1700 的方法从步骤 1702 开始,在该步骤中,命令处理器 1112 接收数据库的标识符。在一个实施例中,数据库的标识符包括文件组的标识符。文件组的标识符可以作为由客户机 1602 的用户向数据库服务器实例 112<sub>1</sub> 提交的命令——诸如 T-SQL 命令——的部分来接收。

[0154] 在步骤 1704 中,命令处理器 1112 接收存储层实例的标识符。存储层实例的标识符可以包括例如已被指派给存储层实例的名称。存储层实例的标识符可以作为由客户机 1102 的用户向数据库服务器实例 112<sub>1</sub> 提交的命令——诸如 T-SQL 命令——的部分来接收。命令可以是用来在步骤 1702 中提供数据库的标识符的相同命令。

[0155] 在步骤 1706 中,数据虚拟化管理器 1612 将来自在步骤 1702 中标识的数据库的数据存储在由步骤 1704 中标识的存储层实例逻辑表示的存储位置 1604。取决于实现,这可以涉及以文件格式或原始存储格式来存储数据。这也涉及将来自数据库的数据存储在被定位在与系统 100 内的不同计算机系统相关联的不同数据存储设备内的存储位置。数据虚拟化管理器 1612 可以通过向在这样的不同的计算机系统上执行的数据虚拟化管理器代理发送命令来执行此功能。取决于不同的因素,此步骤可包括将来自数据库的数据的不同段的克隆存储在不同存储位置里的每一个处,将来自数据库的数据的相同段的克隆存储在这些存储位置里的每一个处,或两者兼有。在一个实施例中,此步骤也可以涉及将来自数据库的数据存储在被定位在同一数据存储设备内的存储位置。

[0156] 下面提供了用于创建数据库以及将存储层与数据库的文件组 / 日志组进行关联的示例命令句法 :

[0157]

```

CREATE DATABASE database_name
    [ ON
        [ PRIMARY ] [ <filegroup_spec>
            [ ,<filegroup> [ ,...n] ]
        [ LOG ON { <filegroup_spec> } ]
    ]
    [ COLLATE collation_name ]
    [ WITH <external_access_option> ]
]
[;]

<filegroup_spec> ::=

{
(
    STORAGETIER = 'storage_tier_name'
        [ , REDUNDANCY_FACTOR = redundancy_factor ]
        [ , INITIALIZESIZE = size [ KB | MB | GB |
TB ] ]
        [ , MAXSIZE = { max_size [ KB | MB | |
GB | TB ] | UNLIMITED } ]
        [ , FILEGROWTH = growth_increment [ KB | MB |
| GB | TB | % ] ]
)
}

```

[0158] 在前述命令中, database\_name(数据库名称)是标识正在被创建的数据库的名称。遵循命令项 PRIMARY(主)的标示为<filegroup\_spec>(文件组\_空间)的文件组规范包括将被指派给数据库的主文件组的存储层实例的名称。遵循命令项 LOG ON(日志记录在)的另一文件组规范包括将被指派给数据库的日志文件组的存储层实例的名称。也可以使用文件组规范来将由”<filegroup>[,...n](文件组)[,...n])”表示的其他用户创建的文件组指派给存储层。

[0159] 如也通过示例命令句法所示出的,文件组规范包括可以对文件或日志组设置的各种属性。这些包括REDUNDANCY\_FACTOR(冗余因子)、INITIALSIZE(初始大小)、MAXSIZE(最大大小)和FILEGROWTH(文件增长)。属性REDUNDANCY\_FACTOR指定要为文件组中所包含的对象的每一个段创建的克隆的数量。属性INITIALSIZE表示由系统在该文件组中创建的用于保存/存储信息的任何文件的初始大小。属性MAXSIZE指定由文件组占用的空间的最

大量,包括由克隆占用的空间。属性 FILEGROWTH 指定文件组中的每一文件增长的增量。

[0160] 下面是用于创建名为 MyDB 的数据库的前述命令句法的使用的一个示例 :

[0161]

```
CREATE DATABASE MyDB
ON PRIMARY
(
    STORAGETIER = 'StData',
    INITIALIZESIZE = 4 MB,
    MAXSIZE = 10 MB,
    FILEGROWTH = 1 MB),
FILEGROUP MyDB_FG1
(
    STORAGETIER = 'StTier1',
    INITIALIZESIZE = 1 MB,
    MAXSIZE = 20 MB,
    FILEGROWTH = 1 MB),
LOG ON
(
    STORAGETIER = 'StLog',
    INITIALIZESIZE = 1 MB,
    MAXSIZE = 10 MB,
    FILEGROWTH = 1 MB);
```

[0162] 这里,数据库 MyDB 的主文件组被指派给存储层类型 data 的系统提供的实例——其名为 StData,数据库 MyDB 的日志文件被指派给存储层类型 log 的系统提供的实例——其名为 StLog,而用户创建文件组 MyDB\_FG1 被指派给类型 data 的用户创建存储层实例——其名为 StTier1(St 层 1)。

[0163] 下面是创建名为 testdb1(测试数据库 1) 的数据库的前述命令句法的使用的另一示例 :

[0164]

```
CREATE DATABASE testdb1
ON PRIMARY (
    STORAGETIER = StData1,
    REDUNDANCY_FACTOR = 3)
LOG ON (
    STORAGETIER = StLog1)
go
```

[0165] 这里,数据库 testdb1 的主文件组将被指派给名为 StData1(St 数据 1) 的用户创建存储层实例,而数据库 testdb1 的日志文件将被指派给名为 StLog1 的用户创建存储层实

例。

[0166] 下面是创建名为 testdb2(测试数据库 2) 的数据库的前述命令句法的使用的再一个示例：

[0167]

```
CREATE DATABASE testdb2  
go
```

[0168] 这里,由于没有显式地指定存储层,因此命令处理器 1112 将把数据库 testdb2 的主文件组指派给存储层类型 data 的默认实例,且将把数据库 testdb2 的日志文件指派给存储层类型 log 的默认实例。在此示例中,在流程图 1700 的步骤 1702 中接收到的数据文件的标识符和在步骤 1704 中接收到的存储层实例的标识符不是通过用户命令提供的,而是替代地由数据库系统 100 本身来提供的。

[0169] 也可以执行流程图 1700 的过程,以根据相关联的存储层来存储系统数据库文件。在此情况下,数据库系统 100 提供系统数据库文件以及相关联的存储层实例两者的标识符,且诸如图 6 的数据虚拟化管理器 604 之类的数据虚拟化管理器将数据库文件存储在由存储层逻辑表示的一个或多个存储位置中。例如,数据库系统 100 指定系统数据库 tempdb 的日志文件与系统所提供的存储层实例 StTempLog 相关联,以及数据虚拟化管理器 604 跨由存储层 StTempLog 逻辑表示的存储位置存储系统数据库 tempdb 的日志文件。

[0170] 6. 向存储层指定策略

[0171] 根据本发明的进一步的实施例,可以结合存储层引入策略,以便给用户提供用于控制不同的对象集在数据库内的存储或放置的能力。根据这样的实施例,可以实现,例如,控制谁可以创建、更改或丢弃存储层,或谁可以将与所创建的数据库相关联的文件存储特定存储层的安全性方案。也可以指定其他策略。

[0172] C. 示例计算机系统实现

[0173] 图 18 描绘了在其上可以执行本发明的各个方面的计算机系统 1800 的示例性实现。计算机系统 1800 旨在以常规个人计算机的形式表示通用计算系统。

[0174] 如图 15 所示,计算机系统 1800 包括处理单元 1802、系统存储器 1804,以及将包括系统存储器 1804 的各种系统组件耦合到处理单元 1802 的总线 1806。系统总线 1806 表示若干类型的总线结构中的任何一种总线结构的一个或多个,包括存储器总线或存储器控制器、外围总线、加速图形端口,以及使用各种总线体系结构中的任何一种的处理器或局部总线。系统存储器 1804 包括只读存储器 (ROM) 1808 和随机存取存储器 (RAM) 1810。基本输入 / 输出系统 1812(BIOS) 存储在 ROM 1808 中。

[0175] 计算机系统 1800 还具有下列驱动器中的一个或多个:用于读写硬盘的硬盘驱动器 1814,用于读写可移动磁盘 1818 的磁盘驱动器 1816,以及用于读写诸如 CD ROM、DVD ROM 之类的可移动光盘 1822,或其他光学介质的光盘驱动器 1820。硬盘驱动器 1814、磁盘驱动器 1816,以及光驱动器 1820 分别通过硬盘驱动器接口 1824、磁盘驱动器接口 1826,以及光学驱动器接口 1828 连接到系统总线 1806。驱动器以及它们相关联的计算机可读介质为服务器计算机提供了对计算机可读指令、数据结构、程序模块,及其他数据的非易失存储器。虽然描述了硬盘、可移动磁盘和可移动光盘,但是,也可以使用诸如闪存卡、数字视频盘、随

机存取存储器 (RAM)、只读存储器 (ROM) 等等之类的其他类型的计算机可读介质来存储数据。

[0176] 数个程序模块可被存储在硬盘、磁盘、光盘、ROM, 或 RAM 上。这些程序包括操作系统 1830、一个或多个应用程序 1832、其他程序模块 1834, 以及程序数据 1836。应用程序 1832 或程序模块 1834 可包括, 例如, 所描述的用于实现数据库服务器实例的逻辑, 以及群集基础结构逻辑实例。应用程序 1832 或程序模块 1834 也可以包括, 例如, 用于实现图 12-15 和 17 中所描绘的流程图的一个或多个步骤的逻辑。如此, 那些图形中所示出的每一步骤都也可以被视为被配置成通过该步骤执行所描述的功能的程序逻辑。

[0177] 用户可以通过诸如键盘 1838 和定点设备 1840 之类的输入设备向计算机 1800 中输入命令和信息。其他输入设备 (未示出) 可以包括麦克风、游戏杆、游戏手柄、圆盘式卫星天线、扫描仪等等。这些及其他输入设备常常通过耦合到总线 1806 的串行端口接口 1842 连接到处理单元 1802, 但是, 也可以通过其他接口, 诸如并行端口、游戏端口、通用串行总线 (USB) 端口, 来进行连接。

[0178] 监视器 1844 或其他类型的显示设备也可以经由诸如视频适配器 1846 之类的接口来连接到系统总线 1806。监视器 1844 被用来呈现协助用户 / 操作员配置和控制计算机 1800 的 GUI。除了监视器之外, 计算机 1800 还可包括其他外围输出设备 (未示出), 如扬声器和打印机。

[0179] 计算机 1800 通过网络接口 1850、调制解调器 1852、或用于通过网络建立通信的其他装置连接到网络 1848 (例如, 诸如因特网之类的 WAN 或 LAN)。调制解调器 1852 (可以是内置的或外置的), 通过串行端口接口 1842 连接到系统总线 1806。

[0180] 如此处所使用的, 术语“计算机程序介质”和“计算机可读介质”被用来泛指诸如与硬盘驱动器 1814 相关联的硬盘、可移动磁盘 1818、可移动光盘 1822, 以及诸如闪存卡、数字视频盘、随机存取存储器 (RAM)、只读存储器 (ROM) 等等之类的其他介质。

[0181] 如上文所指出的, 计算机程序 (包括应用程序 1832 及其他程序模块 1834) 可以存储在硬盘、磁盘、光盘、ROM 或 RAM 上。这样的计算机程序也可以通过网络接口 1850 或串行端口接口 1842 来接收。这样的计算机程序, 当执行时, 使得计算机 1800 能实现此处所讨论的本发明的特征。相应地, 这样的计算机程序表示计算机 1800 的控制器。

[0182] 本发明还涉及包括存储在任何计算机可使用介质上的软件的计算机程序产品。这样的软件, 当在一个或多个数据处理设备中执行时, 使数据处理设备如此处所描述的那样操作。本发明的各实施例使用现在已知的或将来已知的任何计算机可使用或计算机可读介质。计算机可读介质的示例包括, 但不仅限于, 诸如 RAM、硬盘驱动器、软盘、CD ROM、DVD ROM、zip 磁盘、磁带、磁存储设备、光存储设备、MEM (存储器)、基于纳米技术的存储设备等等之类的存储设备。

#### [0183] D. 结论

[0184] 尽管上文描述了本发明的各实施例, 但是, 应该理解, 它们只是作为示例来呈现的, 而不作为限制。那些精通有关技术的人员将理解, 在不偏离如所附权利要求书所定义的本发明的精神和范围的情况下, 可以在形式和细节方面进行各种修改。因此, 本发明的范围不应该受到上述示例性实施例的任一个的限制, 而只应根据下面的权利要求和它们的等效内容进行定义。

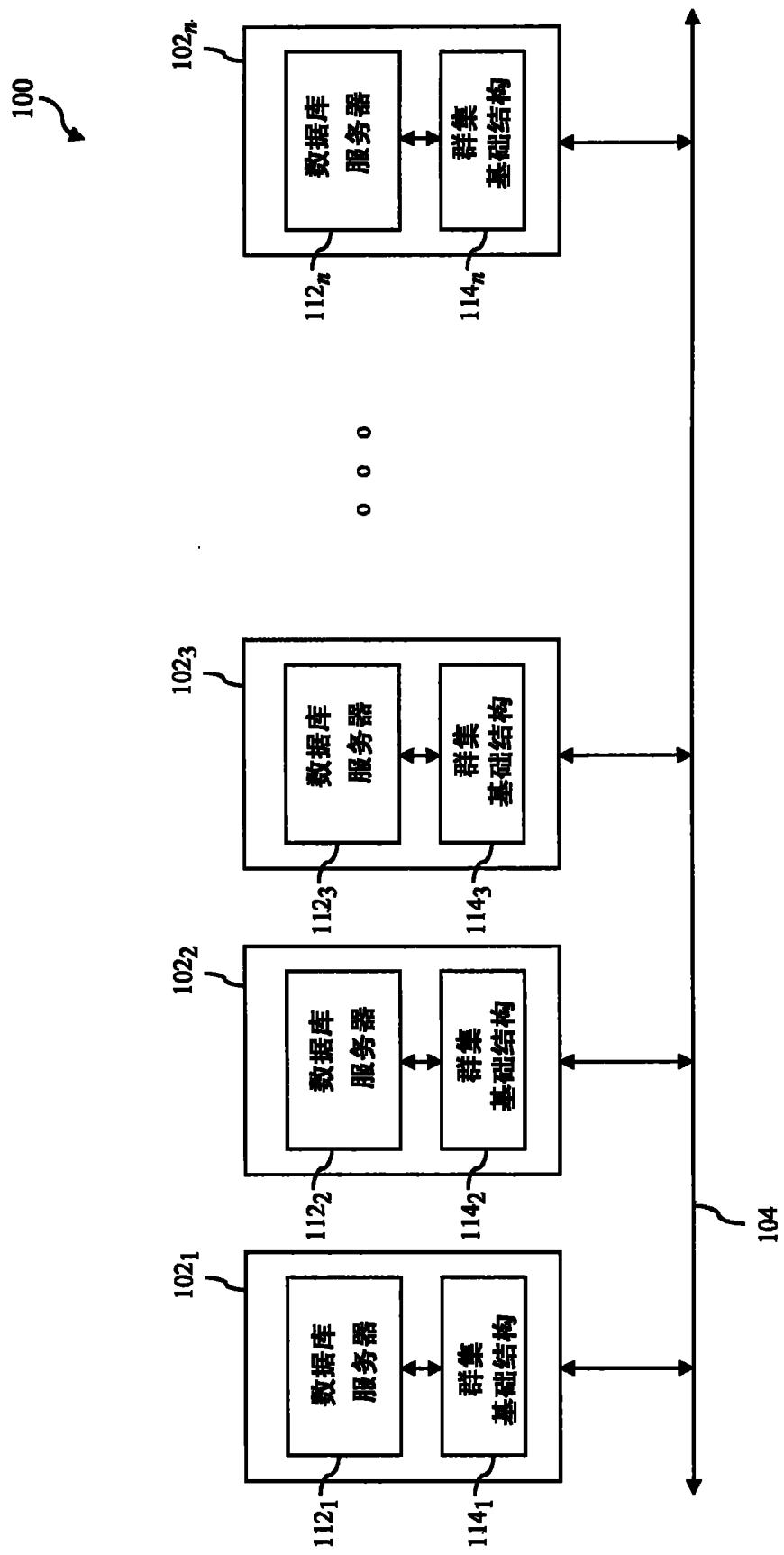


图 1

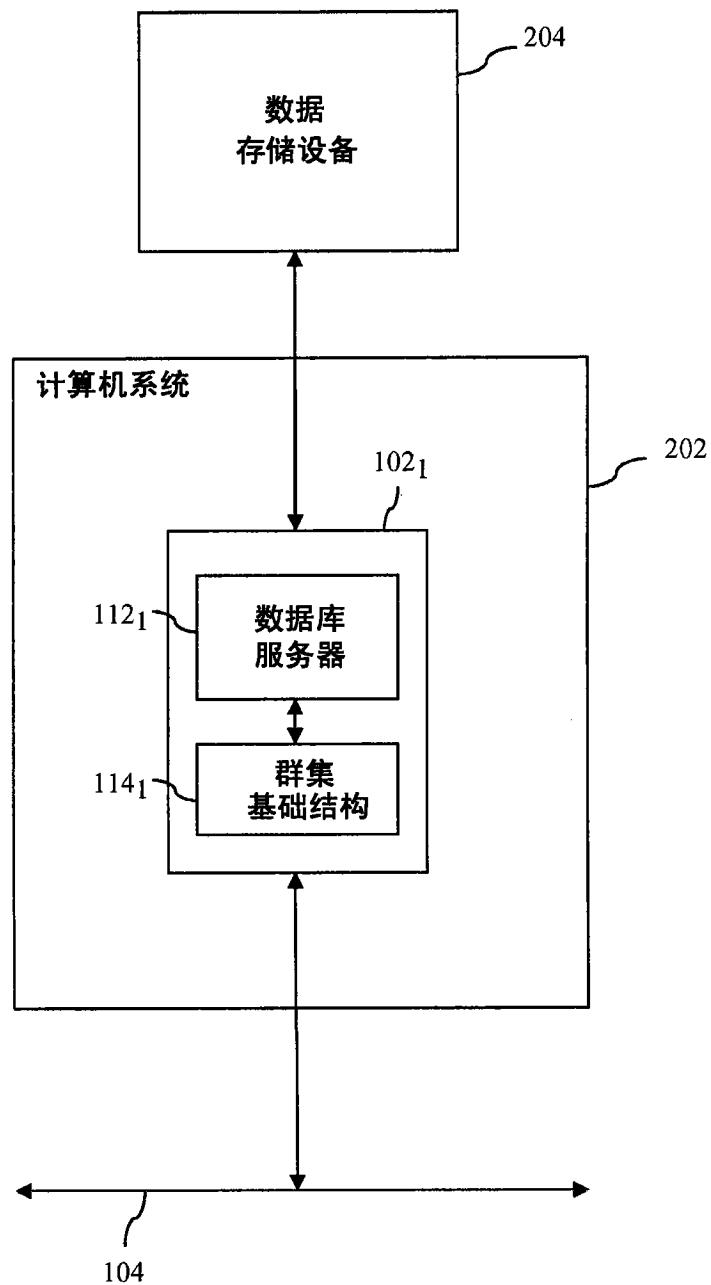


图 2

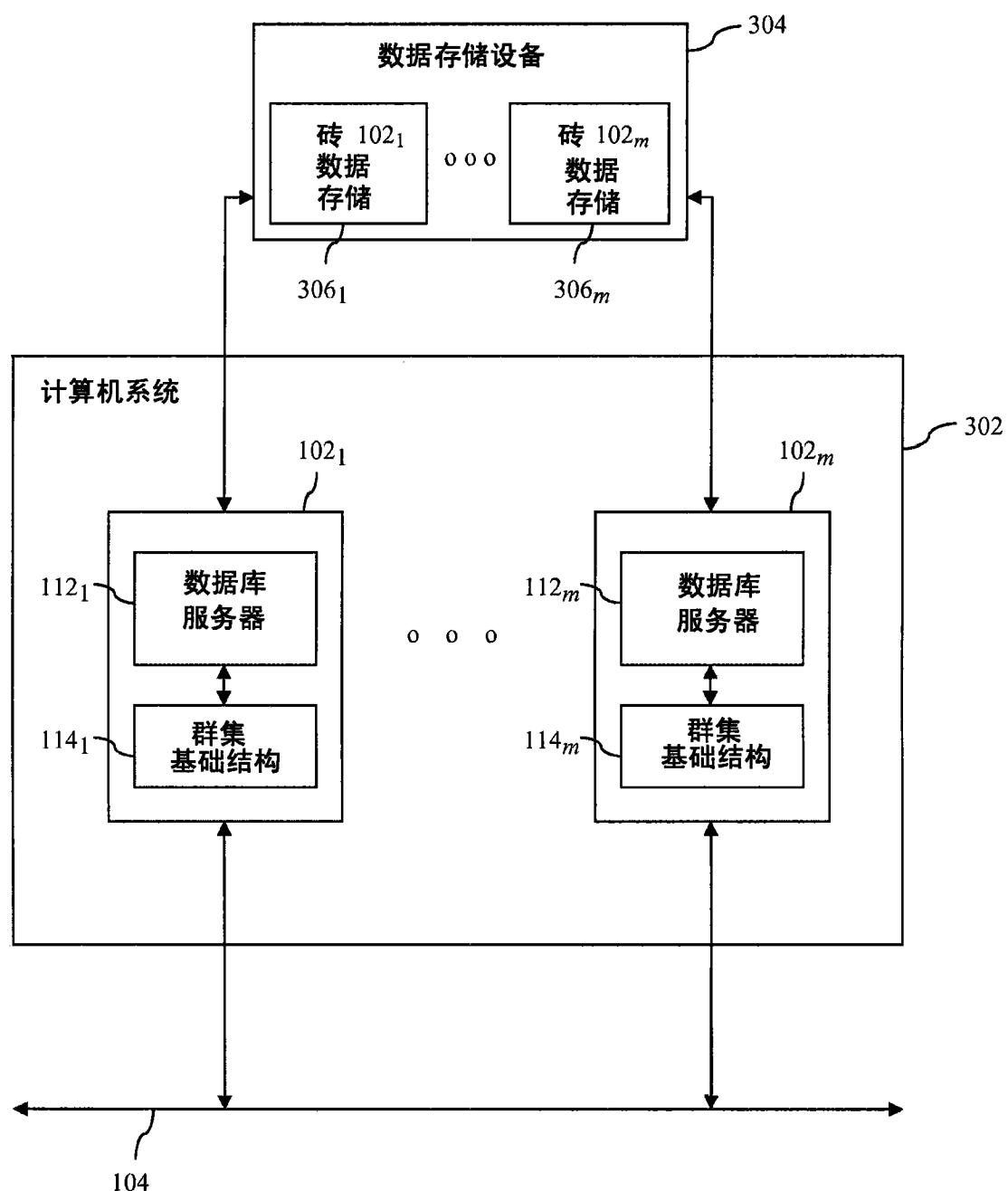


图 3

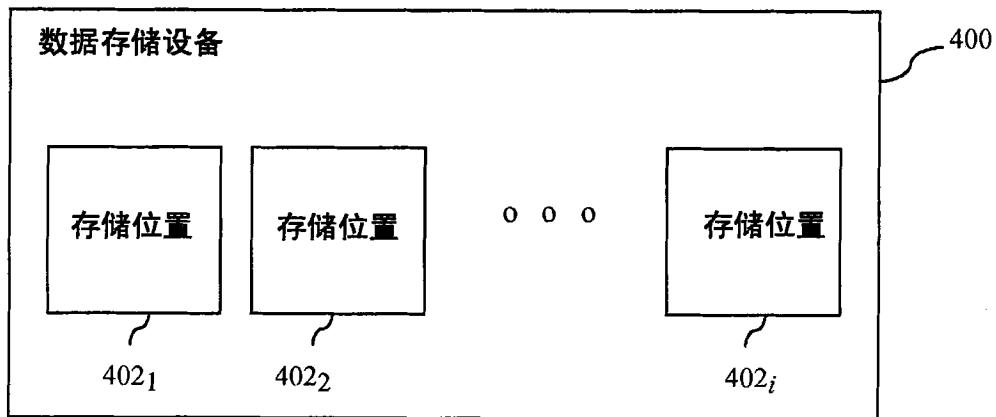


图 4

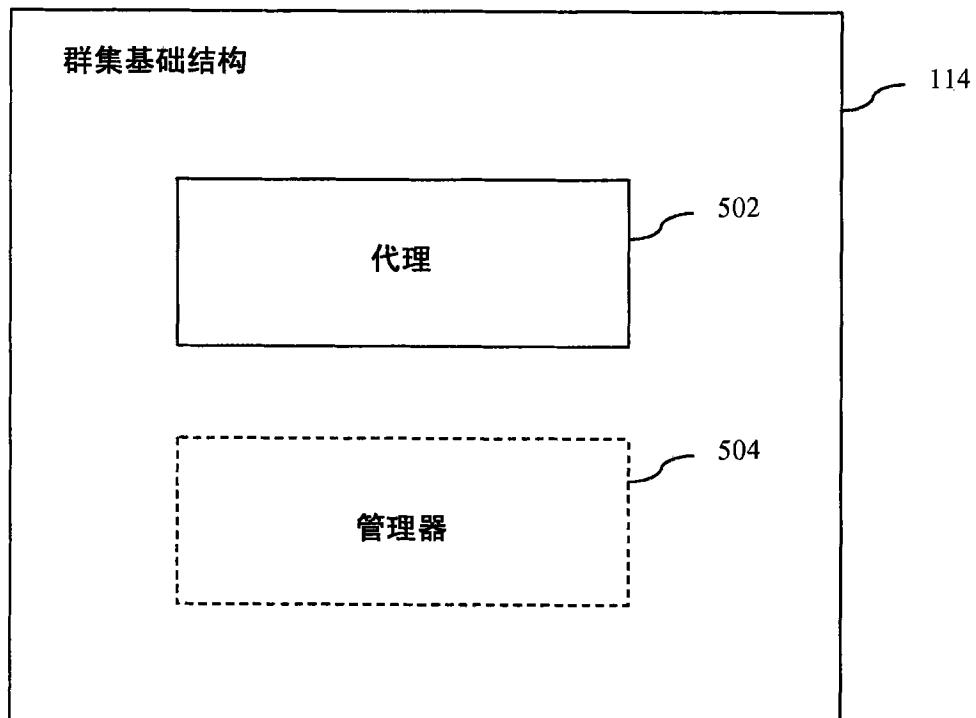


图 5

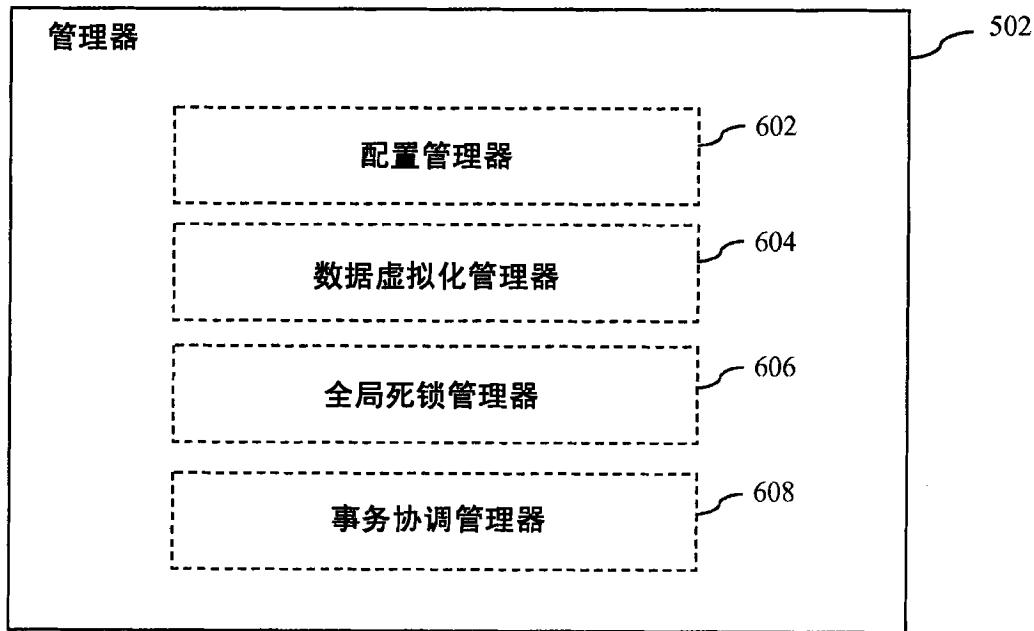


图 6

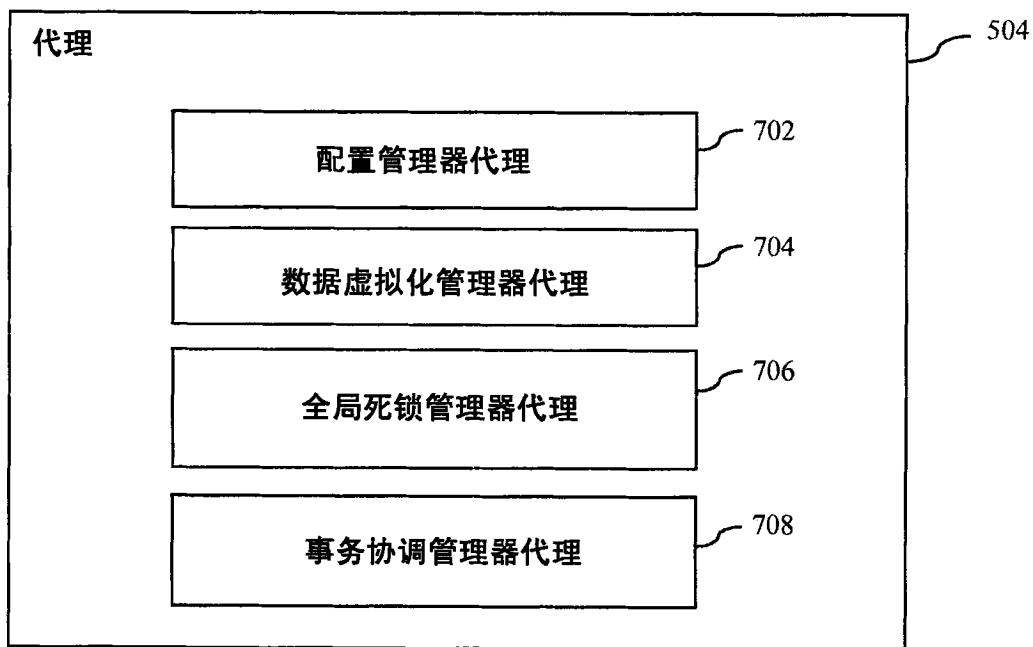


图 7

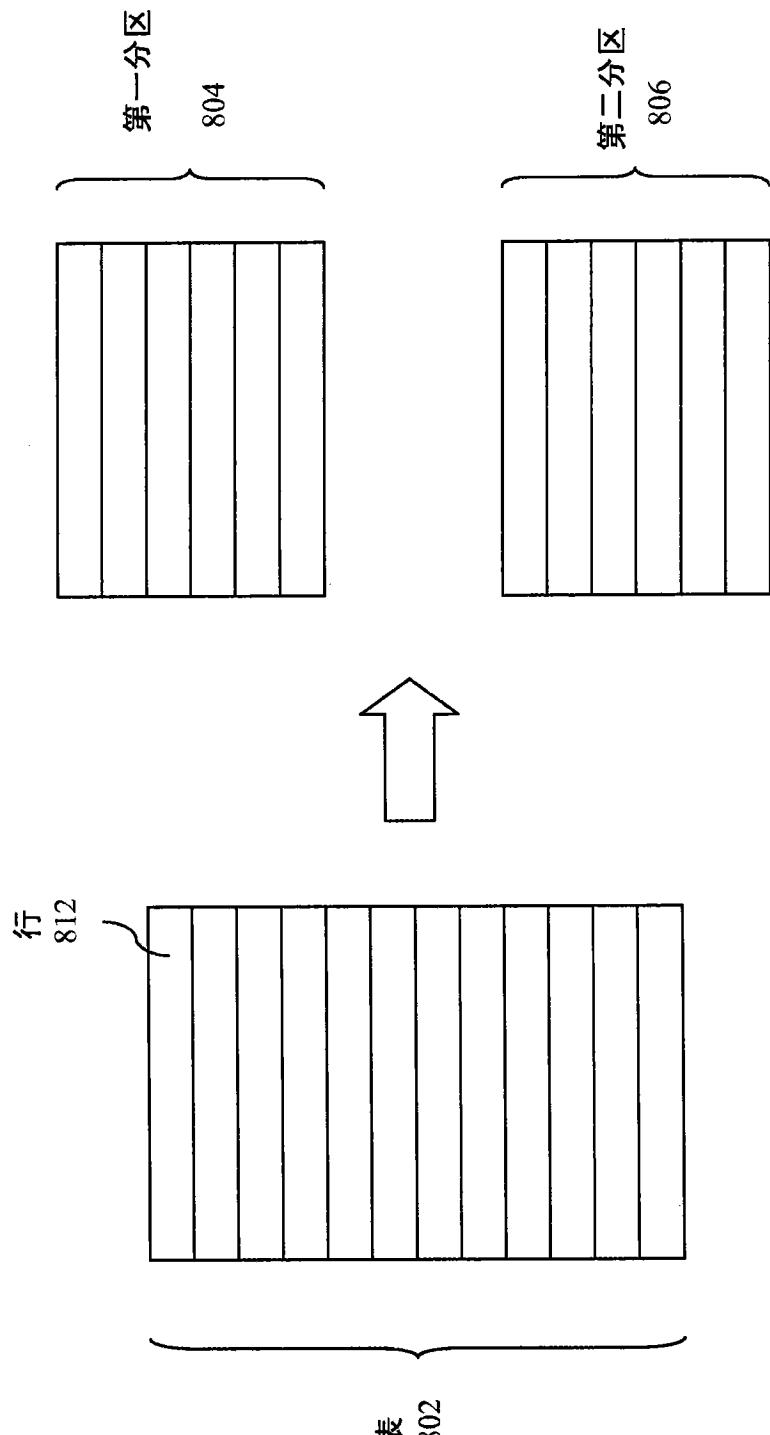


图 8

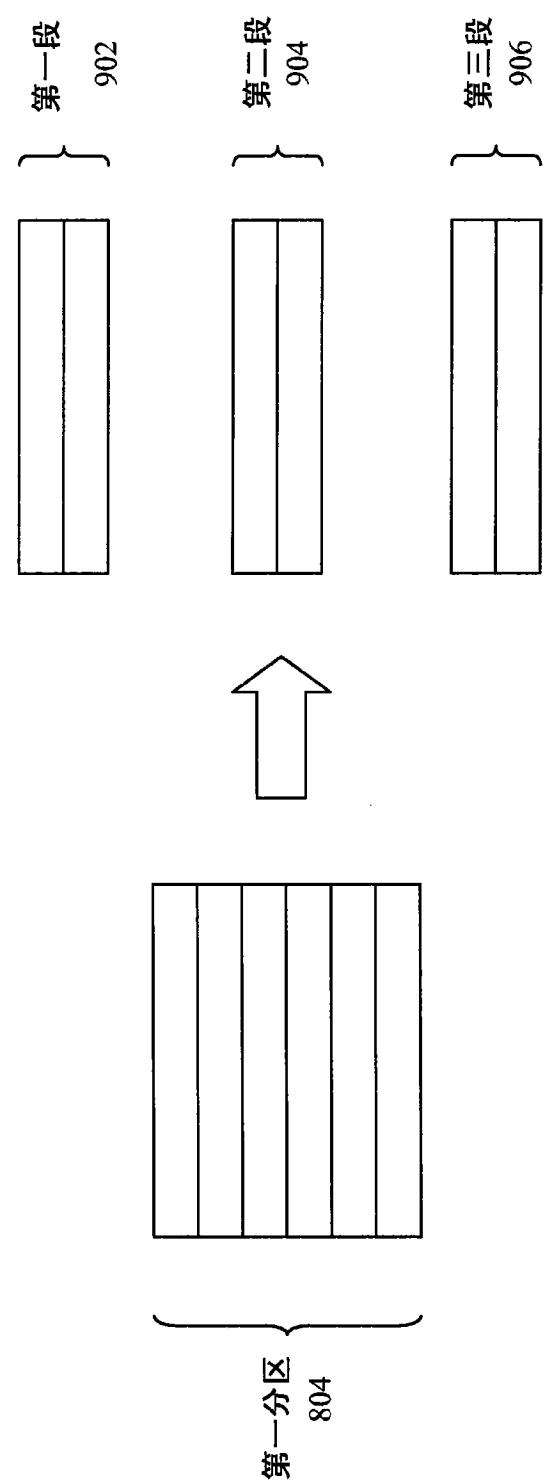


图 9

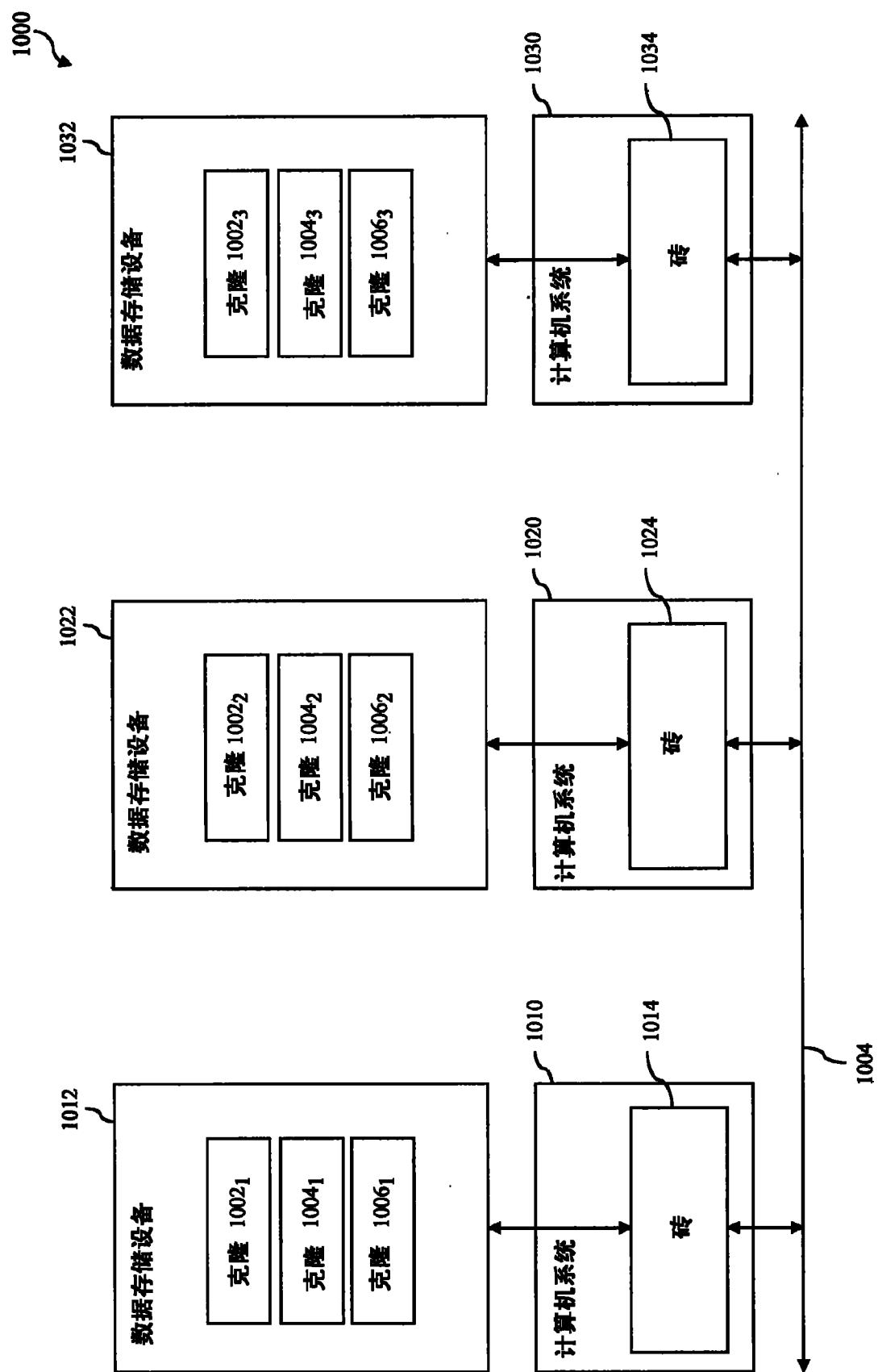


图 10

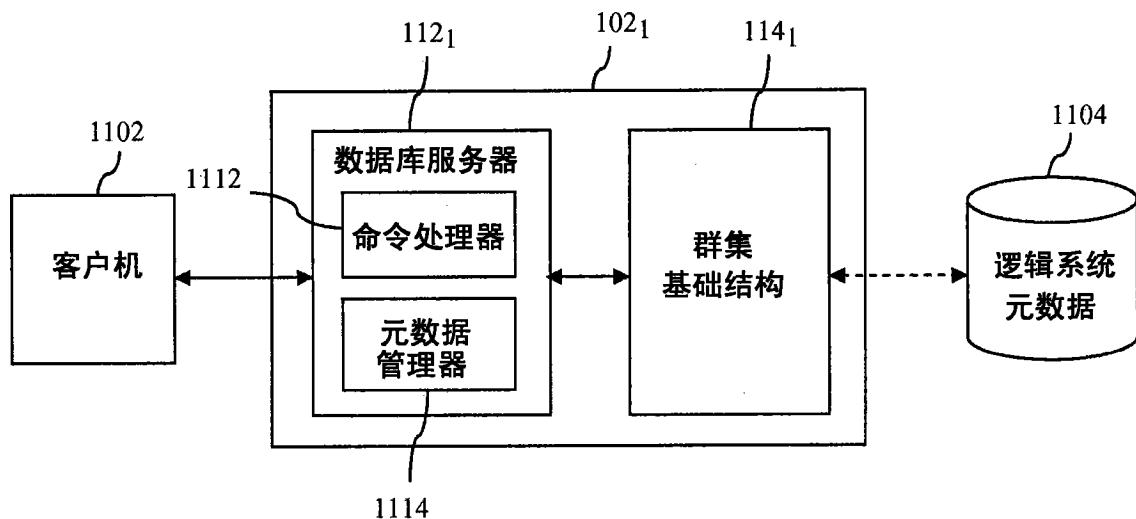


图 11

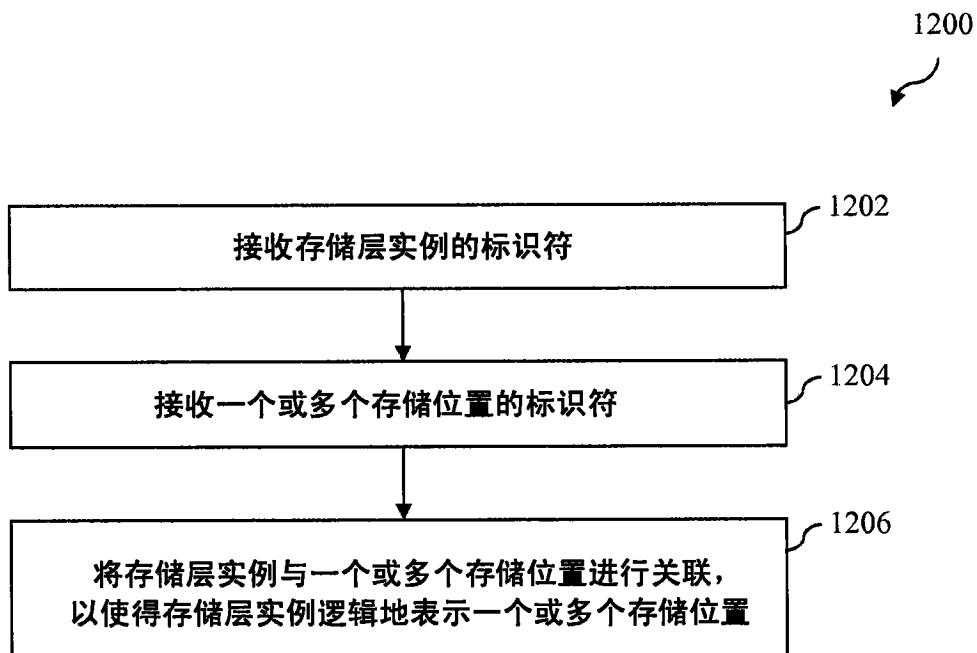


图 12

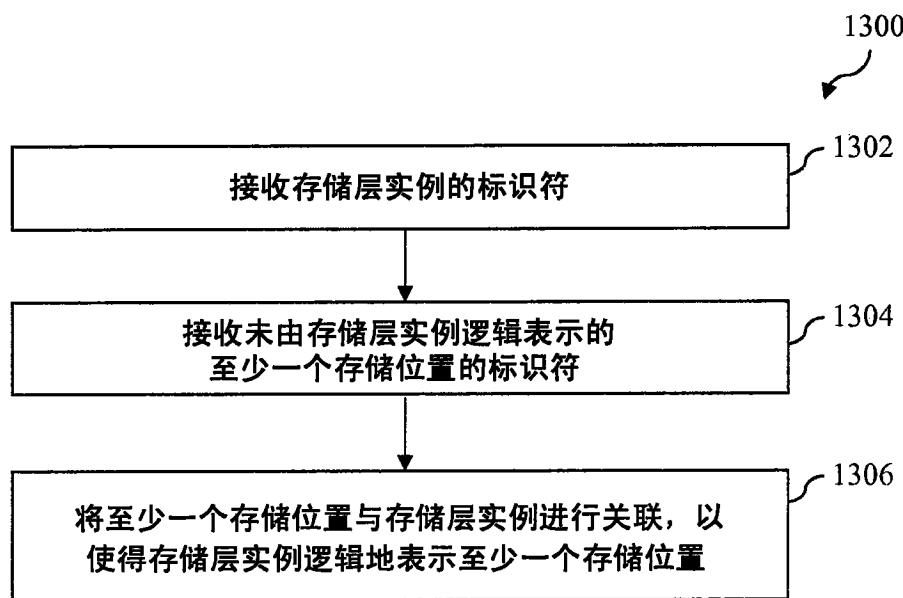


图 13

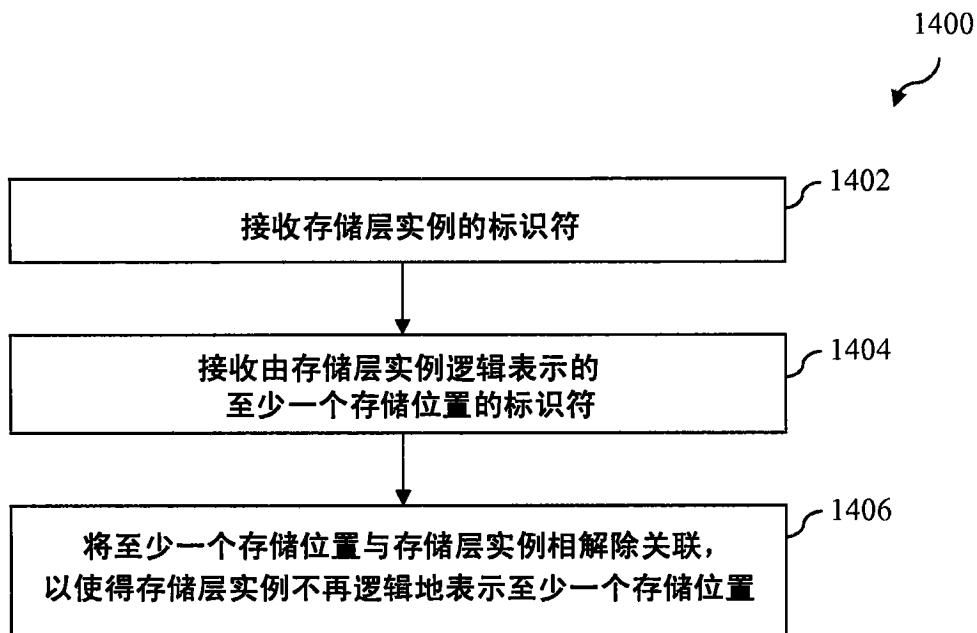


图 14

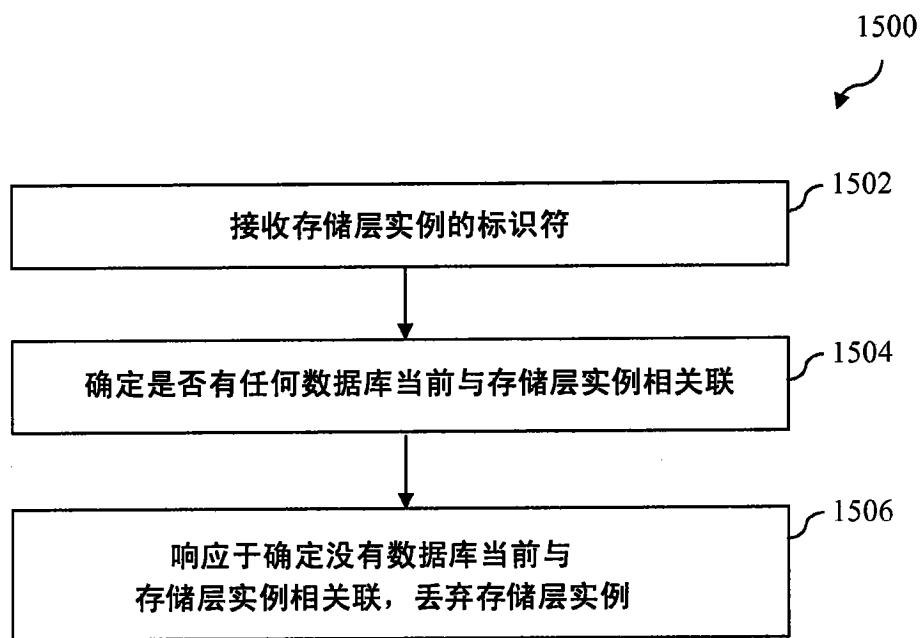


图 15

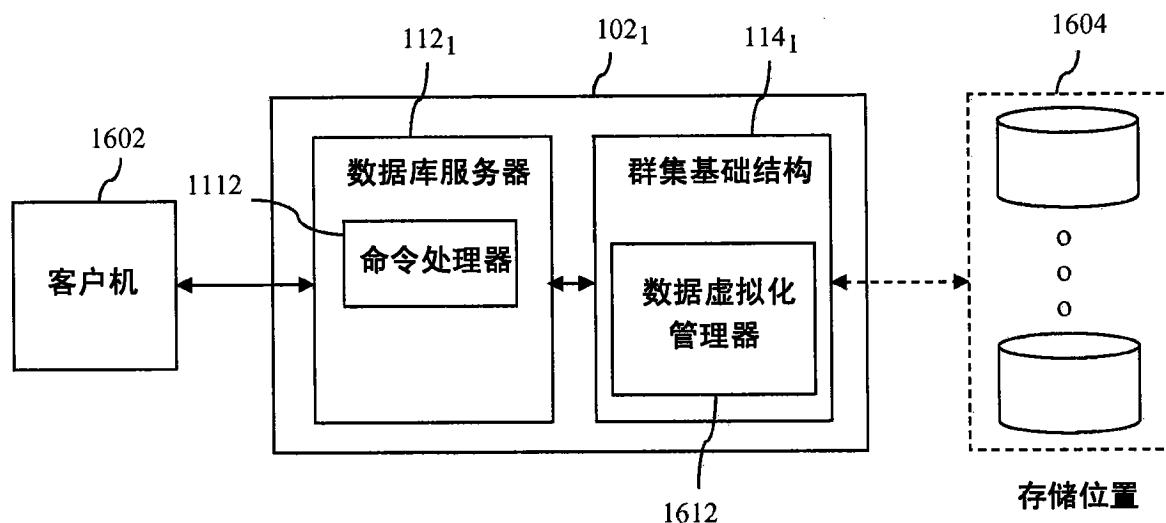


图 16

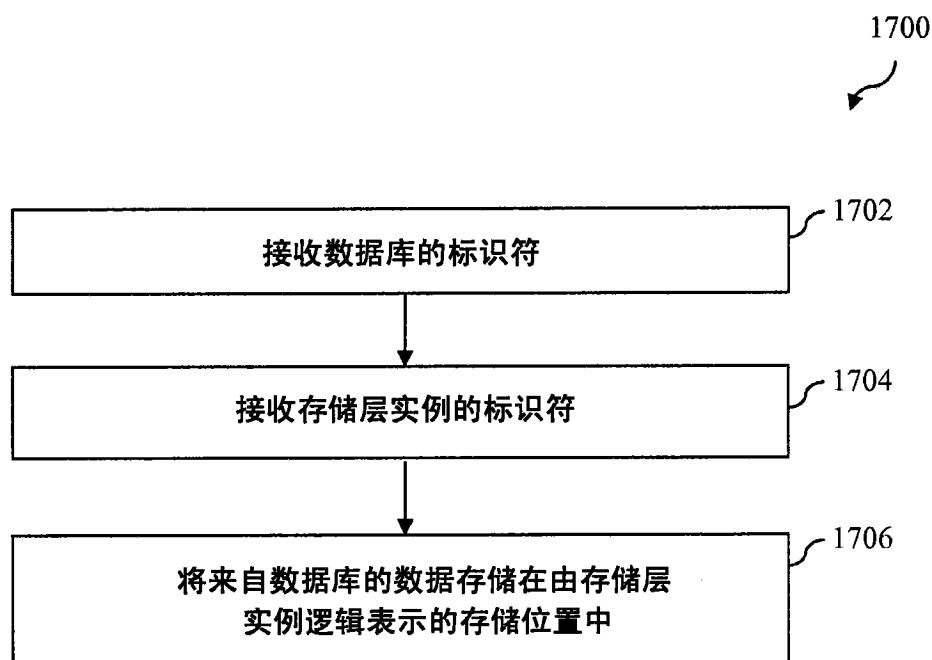


图 17

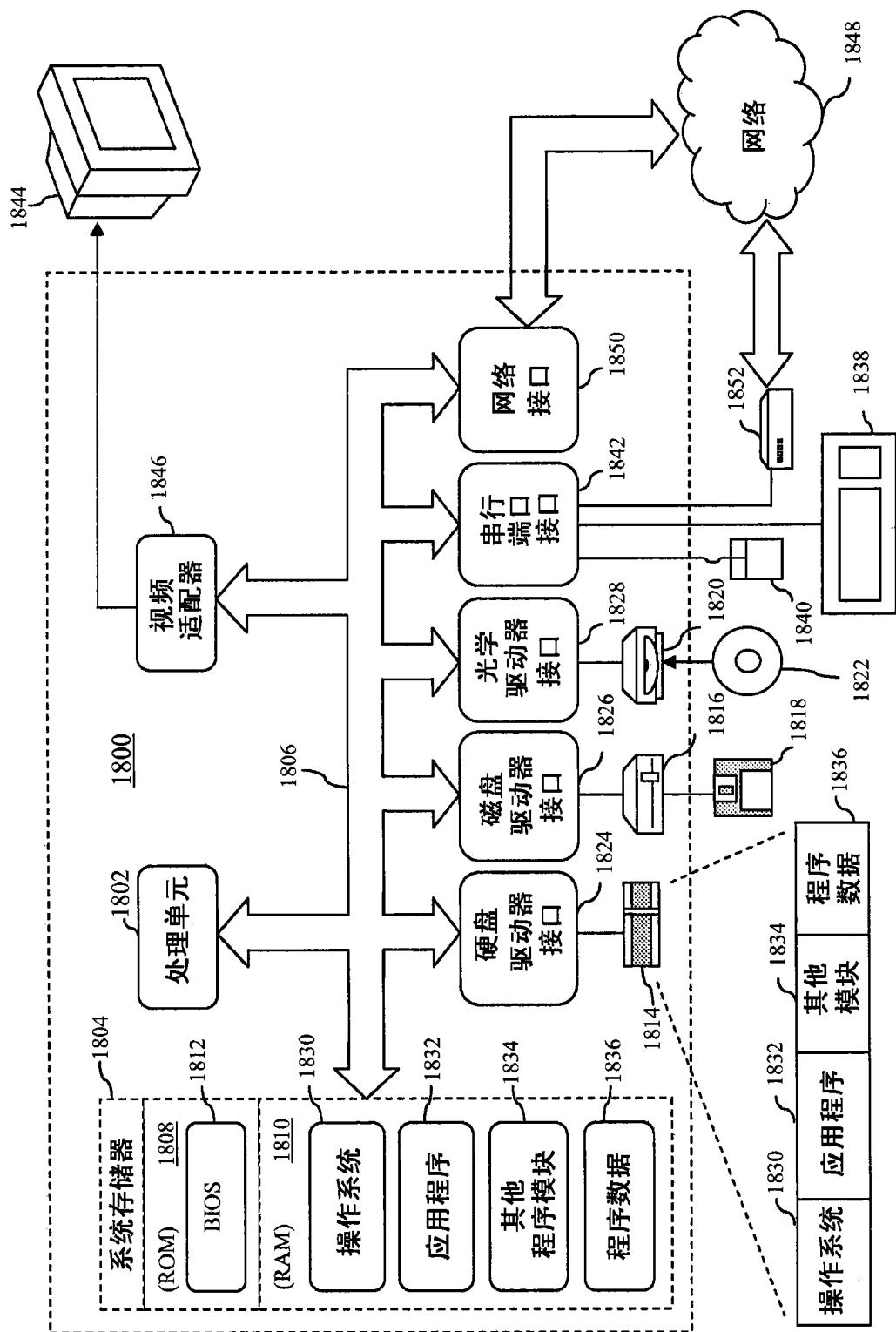


图 18