(19)中华人民共和国国家知识产权局

(12)发明专利

(10)授权公告号 CN 101228523 B
(45)授权公告日 2012.06.06

(21)申请号 200680022927.9

(22)申请日 2006.04.24

(30)优先权数据
60/674,609 2005.04.25 US

(85)PCT申请进入国家阶段日
2007.12.25

(86)PCT申请的申请数据
PCT/US2006/015279 2006.04.24

(87)PCT申请的公布数据
WO2006/116203 EN 2006.11.02

(73)专利权人 网络装置公司
地址 美国加利福尼亚州

(72)发明人 J·A·兰戈 R·M·恩格里斯
P·C·伊斯塔姆 Q·郑
B·M·夸里安 P·格里伊斯
M·B·阿姆杜 K·艾亚尔
R·L·Y·蔡

(74)专利代理机构 中国专利代理(香港)有限公司 72001

代理人 王岳 王忠忠

(51)Int.Cl.
G06F 17/30(2006.01)

(56)对比文件
WO 2004/025429 A2,2004.03.25, 说明书第1页末段-第2页第3段、第5页末段-第21页末段、第23页第3段-第36页第3段和附图3.
US 2003/0158863 A1,2003.08.21, 全文.

审查员 谭李丽

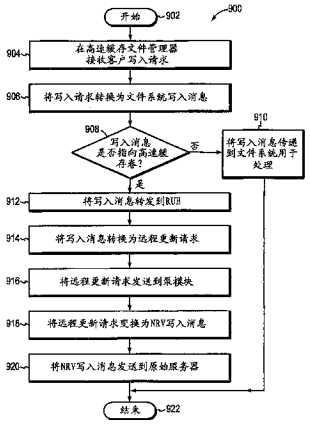
权利要求书 2 页 说明书 17 页 附图 11 页

(54)发明名称

用于高速缓存网络文件系统的系统和方法

(57)摘要

一种网络高速缓存系统，具有连接到原始服务器的多协议高速缓存文件管理器，以提供文件管理器响应于计算机网络上多协议客户端发出的数据访问请求而提供的数据的存储虚拟化。多协议高速缓存文件管理器包括配置为管理稀疏卷的文件系统，该文件系统虚拟化数据的存储空间以由此提供使得能够由多协议客户端访问数据的高速缓存功能。为此，高速缓存文件管理器还包括多协议引擎，该多协议引擎配置为将多协议客户端数据访问请求转换为可由高速缓存文件管理器和原始服务器都可执行的通用文件系统原始操作。



CN 101228523 B

1. 一种网络高速缓存系统，包括：

原始服务器，配置成将数据存储在原始卷；和

耦合到所述原始服务器的高速缓存文件管理器，该高速缓存文件管理器高速缓存存储在所述原始卷的数据，所述高速缓存文件管理器包括：

多协议引擎，配置成将多协议客户端发出的并且由所述高速缓存文件管理器接收的多协议数据访问请求变换为由所述高速缓存文件管理器和原始服务器都可执行的通用文件系统原始操作，以便通过第一协议存储在所述原始卷的信息通过客户端使用第二协议访问所述信息来说是可访问的；

稀疏卷，该稀疏卷通过逻辑地表示所述高速缓存文件管理器的物理存储资源来虚拟化数据的存储空间以由此响应于所述多协议客户端发出的所述多协议数据访问请求而提供由所述高速缓存文件管理器服务的数据的存储虚拟化；和

文件系统，配置成管理所述稀疏卷，其中所述稀疏卷具有从在所述稀疏卷上的文件遗失的数据的至少一个块，并且所述至少一个遗失的块被所述原始服务器存储在所述原始卷。

2. 权利要求 1 所述的网络高速缓存系统，还包括所述高速缓存文件管理器的本地高速缓存，该本地高速缓存包括适于服务由一个或多个客户端从一个或多个存储对象请求的数据的稀疏卷，所述存储对象具有表示从所述稀疏卷中遗失的给定文件的数据的所述至少一个块的至少一个缺少块，其中所述缺少块的遗失数据以不能被客户端看见的方式使用远程获取操作来取得。

3. 权利要求 2 所述的网络高速缓存系统，其中所述存储对象是文件和逻辑单元号之一。

4. 权利要求 2 所述的网络高速缓存系统，其中所述稀疏卷是包括耦合到所述高速缓存文件管理器的一个或多个存储设备的高速缓存卷。

5. 权利要求 4 所述的网络高速缓存系统，还包括所述高速缓存文件管理器的远程更新引擎 (RUE)，该 RUE 配置为将修改所述高速缓存卷的任何文件系统操作转发到所述原始服务器。

6. 权利要求 5 所述的网络高速缓存系统，还包括所述高速缓存文件管理器的截取器，该截取器配置为实现高速缓存弹出策略以在所述高速缓存卷变满时回收存储空间。

7. 一种用于操作网络高速缓存系统的方法，包括：

在高速缓存文件管理器上提供高速缓存卷，其中所述高速缓存卷通过耦合到所述高速缓存文件管理器的原始服务器高速缓存存储在原始卷的数据，所述高速缓存卷是跨越附着到所述高速缓存文件管理器的一个或多个存储设备而存储的稀疏卷，并且该稀疏卷具有在所述原始卷上存储的所述高速缓存卷上的给定文件的数据的至少一个块；

接收指向所述系统的所述高速缓存文件管理器上的存储对象的数据访问请求；

通过所述高速缓存文件管理器的多协议引擎将数据访问请求变换为由所述高速缓存文件管理器和原始服务器都可执行的通用文件系统原始操作，以便通过第一协议存储在原始卷的信息通过客户端使用第二协议访问所述信息来说是可访问的；

确定数据访问请求是否修改在所述高速缓存文件管理器的所述高速缓存卷上存储的数据；

如果是，将所变换的数据访问请求从所述高速缓存文件管理器传送到所述系统的原始服务器；和

在所述原始服务器处理所述变换的数据访问请求。

8. 权利要求 7 所述的方法，还包括：

如果所述数据访问请求没有修改所述高速缓存卷上存储的数据，将该请求传递到所述高速缓存文件管理器的文件系统中；

确定所述请求所请求的数据是否驻留在所述高速缓存文件管理器的本地高速缓存上；

如果未驻留，在所述高速缓存文件管理器上生成一个或多个获取操作以从所述原始服务器取得所请求的数据；和

在从所述原始服务器接收响应时，将获取的数据存储在所述高速缓存文件管理器的所述本地高速缓存中。

9. 权利要求 8 所述的方法，还包括：

如果所请求的数据驻留在所述高速缓存文件管理器的所述本地高速缓存上，在所述高速缓存文件管理器服务所述请求。

用于高速缓存网络文件系统的系统和方法

技术领域

[0001] 本发明涉及高速缓存系统，并且更特别地涉及高速缓存文件管理器响应于计算机网络上多协议客户端发送的数据访问请求而服务的数据的存储虚拟化。

背景技术

[0002] 通常，带有远程位置的机构可能需要复制关键数据，诸如工程应用和程序库到不同的位置。为了使这些关键数据在那些远程位置让用户可得到，而不引起网络延迟，所述机构可消耗大量资源（诸如在文件服务器上执行的文件系统）来管理复杂的复制基础结构和过程。数据复制是已知的技术，其使得可以对通常只读的数据集进行分布式在线访问。传统的数据复制可严重依赖于文件系统镜像法，以创建在分布式服务器上数据集的整体只读副本。

[0003] 由文件系统镜像法生成的镜像通常需要大量的管理开销。例如，管理员必须确定需要复制什么数据，以及为每个镜像管理物理资源（文件系统、文件服务器等）。随着数据集的增长，这类数据复制变得越来越不实用。此外，复制的基础结构可能需要远程位置处存在服务器，以存储复制的数据，因而阻止了机构将它们的服务器基础结构合并到中心位置。因此，存在这样的需要，即消除该高代价的复制基础结构和过程，而不损失立刻访问关键数据的好处。

[0004] 数据复制镜像法的一个替换方式是代理高速缓存。通常，代理高速缓存系统用于按照需要来透明地复制数据集。典型的代理高速缓存系统包括耦合到后端存储系统或具有远程存储器的“原始服务器”的前端存储系统或具有本地存储器的“代理设备”，即“高速缓存”。当高速缓存不能满足客户端请求时，将该请求传递给原始服务器。服务器的响应转而传递回做出请求的客户端并且所有关联的数据高速缓存在本地存储器中。这类事务称为“高速缓存未中”。通常，高速缓存未中导致数据，例如文件系统数据，“填充”到高速缓存中。当需要满足客户请求的数据在高速缓存中可得到时，代理设备可构造且发送响应，而不用与其关联的服务器通信。这种事务称为“高速缓存命中”。使用高速缓存未中事务，代理设备允许客户端修改设备上文件系统的状态。与一般的复制相比，这使得可以进行自动复制，而不约束客户端进行只读访问。

[0005] 传统代理高速缓存解决方案使得可以分布数据，例如文件，到远程位置，而不需要持续地让内行进行管理。这种代理高速缓存解决方案的一个例子在美国专利申请序列号 (P01-1509) 中描述，标题为 Apparatus and Method for a Proxy Cache，申请人为 E.Ackaouy 且转让给 Network Appliance，Inc.，Sunnyvale，California。具有高速缓存的代理存储系统或装置连接到服务器存储系统。文件系统管理代理装置提供的文件集合；客户端使用文件系统协议访问这些文件，例如网络文件系统 (NFS) 和 / 或公共因特网文件系统 (CIFS) 协议。在响应中，代理装置使用基于文件句柄的文件索引散列方法提供文件。

[0006] 概括地叙述，代理装置“倾听”客户端发出的 NFS/CIFS 数据访问请求并且确定它是否可以使用散列方法来本地服务该请求。为此，代理装置在转到其文件系统以进行高速

缓存确定之前将客户端请求转换为唯一的高速缓存名称。在文件句柄上执行的散列函数产生高速缓存名称，文件系统使用该名称以获得高速缓存的文件或者寻找存储标识符以确定文件是否驻留在高速缓存中。如果文件驻留在高速缓存中，则确定客户端请求的所有数据是否都驻留在高速缓存中。如果不是，装置传递请求到服务器。当服务器用请求的数据或应答响应时，装置将服务器响应传递到客户端。代理装置还用服务器的响应“填充”其高速缓存以确保可由装置服务后续的客户端请求。

[0007] 本发明部分针对改进的高速缓冲系统，其使得能够由客户端多协议访问该系统提供的数据。此外，本发明部分针对改进的高速缓冲系统，其使得能够由客户端有效地访问该系统使用文件系统数据结构和名称提供的数据。而且，本发明部分针对改进的高速缓冲系统，其响应于客户端发出的多协议数据访问请求而提供由系统服务的数据的虚拟存储化。在上下文中，存储虚拟化表示将存储的透明视图呈现给这样的客户端，该客户端涉及通常在网络上协同来自多存储系统的存储资源。

发明内容

[0008] 本发明涉及一种具有否合到原始服务器的多协议高速缓冲存储系统（文件管理器）的网络高速缓冲系统，以响应于计算机网络上多协议客户端发出的数据访问请求而提供由该文件管理器服务的数据的存储虚拟化。多协议高速缓存文件管理器包括文件系统，该文件系统配置为管理稀疏的卷，以由此提供使得能够由多协议客户端访问数据的高速缓存功能。为此，高速缓存文件管理器还包括多协议引擎，该引擎配置为将多协议客户端数据访问请求变换为由高速缓存文件管理器和原始服务器都可执行的通用文件系统原始操作。

[0009] 在说明性实施例中，由高速缓存文件管理器的“本地高速缓存”部分地提供了高速缓存功能，所述“本地高速缓存”包括高速缓存卷，该高速缓存卷包括一个或多个耦合到所述高速缓存文件管理器的磁盘。根据本发明的第一方面，高速缓存卷说明性地实现为适于服务客户端从一个或多个例如文件的存储对象请求的数据的稀疏卷，所述存储对象具有从所述高速缓存卷遗失（即没有本地存储在其磁盘上）的至少一个块（即缺少块）。缺少块的遗失数据存储在原始服务器上且以对客户端透明的方式使用远程获取操作被说明性地获取（“填充”）。

[0010] 有利地，本发明利用多协议高速缓存文件管理器的存储空间，使得能够让客户端对网络高速缓存系统服务的数据快且有效地访问。不像之前的高速缓存系统需要显式的文件的句柄到对象的存储转换，新的多协议高速缓存文件管理器使得客户端能够通过使用文件系统且特别地使用由文件系统组织的存储对象（文件）的实际名称来有效地访问由网络高速缓存系统提供的数据。而且，高速缓存文件管理器的文件系统与稀疏的卷协同，以对多协议客户端透明的方式提供对所服务数据的存储空间虚拟化。

附图说明

[0011] 通过结合附图参考下面的描述，可以更好地理解本发明的上述和进一步的优点，附图中相同的附图标记表示同样或功能类似的元件：

[0012] 图 1 是根据本发明的实施例的示例网络环境的示意性框图；

[0013] 图 2 是根据本发明的实施例的示例存储操作系统的示意性框图；

[0014] 图 3 是根据本发明的实施例的示例信息节点 (inode) 的示意性框图；

[0015] 图 4 是根据本发明的实施例的示例缓冲树的示意性框图；

[0016] 图 5 是可有利地用于本发明的文件缓冲树的说明性实施例的示意性框图；

[0017] 图 6 是根据本发明的实施例的示例聚集的示意性框图；

[0018] 图 7 是根据本发明的实施例的示例磁盘上布局的示意性框图；

[0019] 图 8 是根据本发明的实施例的示例 fsinfo 块的示意性框图；

[0020] 图 9 是根据本发明的实施例示出处理数据修改访问请求的过程步骤的流程图；

[0021] 图 10 是根据本发明的实施例示出处理非数据修改访问请求的过程步骤的流程图；

[0022] 图 11 是根据本发明的实施例示出实现高速缓存相干性策略的过程步骤的流程图；和

[0023] 图 12 是根据本发明的实施例示出实现高速缓存弹出策略的过程步骤的流程图。

具体实施方式

[0024] A. 网络环境

[0025] 图 1 是网络高速缓存系统环境 100 的示意性框图，包括前端存储系统，该前端存储系统配置为提供用于服务源自后端存储系统的信息（数据）的高速缓存功能。为此，前端存储系统是说明性地实现为高速缓存文件管理器 120 的计算机，高速缓存文件管理器 120 提供涉及在存储设备，例如磁盘阵列 160 的磁盘 130 上组织信息的存储服务。高速缓存文件管理器 120 包括处理器 122、存储器 124、一个或多个网络适配器 126a、b 和通过系统总线 125 互连的存储适配器 128。高速缓存文件管理器 120 还包括存储操作系统 200，该存储操作系统 200 优选地实现了高级模块，例如文件系统，以便逻辑地将信息组织为磁盘上命名的文件、目录和虚拟磁盘（下文称为特定文件或“块”）存储对象。

[0026] 在说明性实施例中，存储器 124 包括可由处理器和适配器寻址的存储单元以用于存储软件程序代码。存储器的一部分可进一步组织为用于存储与本发明关联的数据结构的缓冲存储器 170。处理器和适配器可转而包括配置为执行软件代码和操纵数据结构的处理元件和 / 或逻辑电路。通常部分驻留在存储器中且由处理元件执行的存储操作系统 200，尤其通过调用文件管理器 120 执行的存储操作来在功能上组织所述文件管理器。对本领域技术人员明显的是，其它处理和存储器装置，包括各种计算机可读介质，可用于存储和执行关于这里描述的发明技术的程序指令。

[0027] 网络适配器 126a、b（下文通常称为“网络适配器 126”）包括需要在计算机网络 140 上将高速缓存文件管理器连接到客户端和后端存储系统的机械、电子和信令电路，所述计算机网络 140 可包括点到点连接或共享介质，例如局域网 (LAN) 或广域网 (WAN)。说明性地，计算机网络 140 可实现为以太网或光纤 (FC) 网络。客户端 110 可通过根据预定义协议，例如传输控制协议 / 网际协议 (TCP/IP) 交换离散帧或数据包来在网络 140 上与文件管理器 120 通信。

[0028] 客户端 110 可以是配置为执行应用 112 的通用计算机。而且，客户端 110 可根据信息递送的客户端 / 服务器模型与高速缓存文件管理器 120 交互。即是，通过在网络 140 上交换包，客户端可请求高速缓存文件管理器的服务，且文件管理器返回客户端所请求服务

的结果。当以文件和目录形式访问信息时，客户端可发送包，该包包括在 TCP/IP 上的基于文件的访问协议，例如公共因特网文件系统 (CIFS) 协议或网络文件系统 (NFS) 协议。可替换地，当以块的形式访问信息时，客户端可发送包，该包包括基于块的访问协议，例如在 TCP 上封装的小型计算机系统接口 (SCSI) 协议 (iSCSI) 和在光纤通道 (FCP) 上封装的 SCSI。

[0029] 存储适配器 128 与在文件管理器 120 上执行的存储操作系统 200 协同以访问用户（或客户端）请求的信息。所述信息可存储在任何类型的可写存储设备媒体的被附着阵列上，所述可写存储设备媒体诸如是录像磁带、光学介质、DVD、磁带、磁泡存储器、电子随机存取存储器、微电子机构和任何其它适于存储信息，包括数据和奇偶信息的类似媒体。然而，如这里所说明性地描述的，所述信息优选地存储在阵列 160 的磁盘 130 上，例如 HDD 和 / 或 DASD。存储适配器包括在 I/O 互连布置，例如传统高性能 FC 串行链路拓扑上耦合到磁盘的输入 / 输出 (I/O) 接口电路。

[0030] 在阵列 160 上的信息存储优选地实现为一个或多个存储“卷”，该卷包括物理存储磁盘 130 的集合，所述物理存储磁盘 130 协同以定义一个或多个卷上的卷块号 (vbn) 空间的整体逻辑布置。尽管不是必须的，通常每个逻辑卷与其自己的文件系统相关联。通常，逻辑卷 / 文件系统中的磁盘组织为一个或多个组，其中每个组可作为独立磁盘冗余阵列 (RAID) 来操作。大多数 RAID 实现，例如 RAID-4 级实现，通过在 RAID 组中跨越给定数目的物理磁盘冗余写入数据带区且对于带区的数据适当地存储奇偶信息来增强数据存储的可靠性 / 完整性。RAID 实现的一个说明性例子是 RAID-4 级实现，尽管应当理解，可根据这里描述的发明原理使用 RAID 实现的其它类型和级别。

[0031] 在一个说明性实施例中，高速缓存文件管理器 120 的高速缓存功能部分地由“本地高速缓存”提供。关于这点，本地高速缓存表示高速缓冲存储器的层次结构，包括 (i) 高级处理器高速缓存 123，(ii) 中级缓冲存储器 170，和 (iii) 包括耦合到文件管理器的一个或多个磁盘 130 的低级“第三”高速缓存卷 150。根据这里进一步描述的本发明的一个方面，高速缓存卷 150 说明性地实现为适于服务客户端 110 从一个或多个例如文件的存储对象请求的数据的稀疏卷，所述存储对象具有从所述高速缓存卷 150 遗失（即没有本地存储在其磁盘上）的至少一个块（即缺少块）。缺少块的遗失数据存储在后端存储系统上且以对客户端透明的方式使用远程获取操作被说明性地取得（“填充”）。

[0032] 后端存储系统是说明性地实现为原始服务器 180 的计算机，像高速缓存文件管理器 120 一样，其提供涉及在组织为原始卷 185 的磁盘上组织信息的存储服务。原始服务器 180 在网络 140 上与高速缓存文件管理器 120 操作地互连，且通常包括类似于文件管理器 120 的硬件。然而，可替换地，原始服务器 180 可执行修改的存储操作系统，该存储操作系统使存储系统适用于原始服务器。在这里进一步描述的可替换实施例中，在网络高速缓存系统环境 100 中可有多个耦合到原始服务器 180 的高速缓存文件管理器 120。

[0033] B. 存储操作系统

[0034] 为了便于访问磁盘 130，存储操作系统 200 实现随处可写的文件系统，该文件系统与虚拟化模块协同以管理高速缓存（稀疏）卷 150 并且“虚拟化”由磁盘 130 提供的存储空间。文件系统逻辑地将信息组织为磁盘上的命名的目录和文件的层次结构。每个磁盘上文件可实现为磁盘块的集合，所述磁盘块配置为存储信息，例如数据，而目录可实现为特定格式的文件，其中存储了名称和到其它文件和目录的链接。虚拟化模块允许文件系统进一

步逻辑地将信息组织为作为命名逻辑单元号 (lun) 输出的磁盘上块的层次结构。

[0035] 在说明性实施例中，存储操作系统优选地是可从 NetworkAppliance，Inc.，Sunnyvale，California 得到的 NetApp ® DataONTAP™ 操作系统，其实现了随处可写文件布局 (WAFL™) 文件系统。然而，明显可预期的是，根据这里描述的发明原理，可增强任何适当的存储操作系统以便于使用。同样地，这里使用的术语“WAFL”应当被用来广义地指任何另外适用于本发明的教导的文件系统。

[0036] 图 2 是可有利地用于本发明的存储操作系统 200 的示意性框图。存储操作系统包括一系列软件层，该软件层组织为形成完整的网络协议栈，或更一般地，形成使用块和文件访问协议为多协议客户端提供数据通路以访问存储在高速缓存文件管理器上的信息的多协议引擎。协议栈包括网络驱动器（例如千兆以太网驱动器）的媒体访问层 210，该媒体访问层 210 对接到网络协议层，例如 IP 层 212 和它的支持传输机制、TCP 层 214 和用户数据报协议 (UDP) 层 216。文件系统协议层提供多协议文件访问，并且为此，包括对直接存储文件系统 (DAFS) 协议 218、NFS 协议 220、CIFS 协议 222 和超文本传输协议 (HTTP) 协议 224 的支持。如 DAFS 协议 218 所需要的，VI 层 226 实现了 VI 体系结构以提供直接访问传输 (DAT) 能力，例如 RDMA。

[0037] iSCSI 驱动器层 228 提供了在 TCP/IP 网络协议层上的块协议访问，而 FC 驱动器层 230 从 / 向高速缓存文件管理器接收 / 发送块访问请求 / 响应。当访问文件管理器上的块时，FC 和 iSCSI 驱动器提供对块的 FC 专用和 iSCSI 专用的访问控制，并且因此管理 lun 到 iSCSI 或 FCP 的输出，或可替换地到 iSCSI 和 FCP 两者的输出。此外，存储操作系统包括实现为 RAID 系统 240 的存储模块和实现磁盘访问协议，例如 SCSI 协议的磁盘驱动器系统 250，所述存储模块根据 I/O 操作来管理向 / 从卷 / 磁盘存储 / 获取信息。

[0038] 存储操作系统 200 还包括网络应用远程卷 (NRV) 协议层 295，该协议层 295 与文件系统 280 对接。通常，NRV 协议用于远程获取没有本地存储在磁盘上的数据块。然而，如这里描述的，NRV 协议可进一步用于高速缓存文件管理器到原始服务器的通信，以根据本发明的原理获取稀疏高速缓存卷中的缺少块。应当注意，在可替换实施例中，传统文件 / 块级别协议，例如 NFS 协议或其它专有块获取协议可用于代替本发明教导中的 NRV 协议。

[0039] 如这里进一步描述的，存储操作系统 200 的需求生成器 296 用于系统地获取没有本地存储在磁盘，即高速缓存文件管理器 120 的高速缓存卷 150 上的数据块，而泵模块 298 可用于调整从原始服务器 180 请求的那些和其它数据块的获取。而且，根据本发明，截取器 294 实现了高速缓存弹出策略以便在本地高速缓存（例如高速缓存卷 150）变满时回收存储空间，并且远程更新引擎 (RUE292) 用于将任何修改高速缓存卷 150 的文件系统操作转发到原始服务器 180。尽管这里示出和描述为单独的软件模块，可替换地，需求生成器 296、泵 298、截取器 294 和 RUE292 可集成在操作系统 200 的单一模块内。而且，应当注意这些模块可以实现为硬件、软件、固件或其任意组合。

[0040] 桥接磁盘软件层和多协议引擎层的是文件系统 280 实现的虚拟化系统，该虚拟化系统与说明性地实现为例如 vdisk 模块 290 和 SCSI 目标模块 270 的虚拟化模块交互。vdisk 模块 290 在文件系统 280 上层，能够由例如用户接口 (UI) 275 的管理接口访问，以响应用户（例如系统管理员）向文件管理器发送命令。UI275 以使得管理员或用户能够访问不同层和系统的方式布置于存储操作系统之上。SCSI 目标模块 270 布置于 FC 和 iSCSI 驱动器 228、

230和文件系统280之间，以提供在块(lun)空间和文件系统空间之间的虚拟化系统的变换层，其中lun表示为块。

[0041] 说明性地，文件系统是基于消息的系统，该系统提供逻辑卷管理能力，用于访问存储在存储设备，例如磁盘上的信息。即是，除了提供文件系统语义，文件系统280还提供通常与卷管理器关联的功能。这些功能包括(i)磁盘聚集，(ii)磁盘的存储带宽的聚集，和(iii)可靠性保证，例如镜像法和/或奇偶性(RAID)。说明性地，文件系统280实现了具有磁盘上格式表示的WAFL文件系统(下文一般称为“随处可写文件系统”)，所述格式表示是基于块的，使用例如4千字节(KB)块且使用索引节点(“信息节点”)以标识文件和文件属性(诸如创建时间、访问许可、大小和块位置)。文件系统使用文件存储元数据，该元数据描述文件系统的布局；这些元数据文件尤其包括信息节点文件。文件句柄，即包括信息节点号的标识符，用于从磁盘获取信息节点。

[0042] 概括地叙述，随处可写文件系统的所有信息节点组织为信息节点文件。文件系统(fs)信息块说明了文件系统中信息的布局且包括文件的信息节点，所述文件包括文件系统的所有其它信息节点。每个逻辑卷(文件系统)具有优选地存储在例如RAID组内的固定位置的fsinfo块。根fsinfo块的信息节点可直接引用(指向)信息节点文件的块，或可间接引用信息节点文件的块，该信息节点文件的块转而引用信息节点文件的直接块。在信息节点文件的每个直接块内是嵌入的信息节点，每个嵌入的信息节点可引用间接块，该间接块转而引用文件的数据块。

[0043] 可操作地，来自客户端110的请求作为计算机网络140上的包转发且到达高速缓存文件管理器120上，其中在网络适配器126处接收包。(层210或层230的)网络驱动器处理所述包，并且如果适当，将其传递到网络协议和文件访问层，以在转发到随处可写文件系统280之前做另外的处理。如这里进一步描述的，如果请求修改了存储在高速缓存卷150上的数据，高速缓存文件管理器120经由NRV写入请求将所述请求传送到原始服务器180。然而，如果请求不修改卷150上的数据，直接将请求传递到文件系统280中，该文件系统280尝试服务所述请求。如果数据未驻留在本地高速缓存上(导致“高速缓存未中”)，高速缓存文件管理器将NRV读取请求发送到原始服务器180，以获取遗失数据。在从服务器180接收响应后，高速缓存文件管理器将获取的数据存储在其本地高速缓存中，用所请求的数据构造回复且将该回复返回到客户端110。

[0044] 然而，如果所请求的数据驻留在本地高速缓存中，高速缓存文件管理器(文件系统280)服务该请求。为此，如果所请求的数据未驻留“在核心”即在缓冲存储器170中，文件系统生成操作以从磁盘130装载(获取)所请求的数据。说明性地，该操作可实现为文件系统280的Load_Block()函数284。如果信息不在高速缓冲器170中，文件系统280使用信息节点号在信息节点文件中检索，以访问适当的条目且获取逻辑vbn。文件系统接着将包括逻辑vbn的消息结构传递到RAID系统240；逻辑vbn将映射到磁盘标识符和磁盘块号(磁盘，dbn)且发送到磁盘驱动器系统250的适当驱动器(例如SCSI)。磁盘驱动器从特定磁盘130访问dbn且将所请求的数据块装载到缓冲存储器170中以便由文件管理器处理。在完成了请求后，文件管理器(和操作系统)在网络140上将回复返回到客户端110。

[0045] 通常，文件系统280提供Load_Block()函数284以从磁盘获取一个或多个块。可响应于读取请求或指向例如文件的示例提前读取算法获取这些块。如这里进一步描述的，

如果文件的缓冲树内的任何所请求的块包含专用 ABSENT 值（由此表示缺少块），则 Load_Block() 函数 284 启动获取操作以使用说明性 NRV 协议 295 从适当的后备存储器（例如原始服务器 180）获取缺少块。一旦获取了块（包括任何数据块），Load_Block() 函数 284 返回所请求的数据。在上面提到的美国专利申请中进一步描述了 NRV 协议，标题为 Architecture forSupporting of Sparse Volumes，申请人为 Jason Lango 等。然而，应当注意，任何其它适合的可从远程后备存储器获取数据的基于文件或块的协议，例如包括 NFS 协议，可有利地用于本发明。说明性地，文件系统还包括首次访问文件时获取信息节点和文件结构的 Load_Inode() 函数 288。

[0046] 应当进一步注意，可替换地，上述需要为在高速缓存文件管理器接收的客户端请求执行数据存储访问的通过存储操作系统层的软件路径可用硬件实现。即在本发明的可替换实施例中，存储访问请求数据路径可实现为逻辑电路，该逻辑电路用现场可编程门阵列 (FPGA) 或专用集成电路 (ASIC) 实现。这类硬件实现增加了文件管理器 120 响应于客户端 110 发送的请求而提供的存储服务的性能。而且，在本发明的另一可替换实施例中，适配器 126、128 的处理元件可配置为各自地卸载来自处理器 122 的一些或所有包处理和存储访问操作，以由此增加文件管理器提供的存储服务的性能。明显可预期的是，这里描述的各种处理、体系结构和过程可用硬件、固件或软件实现。

[0047] 如这里使用的，术语“存储操作系统”一般指可操作执行存储系统中存储功能的计算机可执行代码，所述存储功能例如管理数据访问和在高速缓存文件管理器的情况下，可实现文件系统语义。在这个意义上，ONTAP 软件是这种存储操作系统的一个例子，该软件实现为微内核且包括 WAFL 层，以实现 WAFL 文件系统语义并管理数据访问。存储操作系统还可实现为在通用操作系统，例如 UNIX ® 或 WindowsNT ® 上运行的应用程序，或实现为具有可配置功能性的通用操作系统，该通用操作系统被配置用于这里描述的存储应用。

[0048] 此外，本领域技术人员将理解的是，这里描述的发明系统和方法可应用于任何类型的专用（例如文件服务器、文件管理器或多协议存储装置）或通用计算机，包括实现为或包括存储系统的独立计算机或其一部分。可有利地用于本发明的多协议存储装置的一个例子在美国专利申请序列号 10/215,917 中描述，标题为 Multi-protocolStorage Appliance that Provides Integrated Support for Fileand Block Access Protocols，于 2002 年 8 月 8 日提出。而且，本发明的教导可适用于多种存储系统体系结构，包括但不限于附加到网络的存储环境、存储区域网络和直接附加到客户端或主机的磁盘部件。因此，术语“存储系统”应当广义地包括这种布置以及配置为执行存储功能且与其它设备或系统相关联的任何子系统。

[0049] C. 文件系统组织

[0050] 在说明性的实施例中，文件在随处可写文件系统中表示为适于在磁盘 130 上存储的信息节点数据结构。图 3 是信息节点 300 的示意性框图，其优选地包括元数据段 310 和数据段 350。存储在每个信息节点 300 的元数据段 310 中的信息描述了文件，并且如所示的，包括文件类型（例如普通、目录、虚拟磁盘）312、文件大小 314、文件的时间戳（例如访问和 / 或修改时间）316 和文件的所有权，即用户标识符 (UID 318) 和组 ID(GID 320)。然而，每个信息节点的数据段 350 的内容可根据类型字段 312 内定义的文件（信息节点）类型而不同地解释。例如，目录信息节点的数据段 350 包含文件系统控制的元数据，而普通信

息节点的数据段包含文件系统数据。在后一种情况中，数据段 350 包括与文件相关联的数据的表示。

[0051] 特定地，普通磁盘上信息节点的数据段 350 可包括文件系统数据或指针，后者引用磁盘上用于存储文件系统数据的 4KB 的数据块。优选地，每个指针是逻辑 vbn，以在访问磁盘上的数据时提高文件系统和 RAID 系统 240 之间的效率。给定信息节点的有限大小（例如 128 字节），大小小于或等于 64 字节的文件系统数据全部表示在该信息节点的数据段内。然而，如果文件系统数据大于 64 字节，但小于或等于 64kB，则信息节点（例如第一级信息节点）的数据段包括达 16 个指针，每个指针引用磁盘上的 4kB 的数据块。

[0052] 而且，如果数据的大小大于 64kB 但小于或等于 64 兆字节 (MB)，则信息节点（例如第二级信息节点）的数据段 350 中的每个指针引用间接块（例如第一级块），该间接块包含达 1024 个指针，每个指针引用磁盘上 4kB 的数据块。对于具有超过 64MB 大小的文件系统数据，信息节点（例如第三级信息节点）的数据段 350 中的每个指针引用双重间接块（例如第二级块），该双重间接块包括达 1024 个指针，每个指针引用一个间接（例如第一级）块。转而，间接块包含 1024 个指针，每个指针引用磁盘上的 4kB 的数据块。当访问文件时，文件的每个块可从磁盘 130 装载到缓冲存储器 170 中。

[0053] 当磁盘上信息节点（或块）从磁盘 130 装载到缓冲存储器 170 中时，该磁盘上信息节点（或块）在核心中的对应结构嵌入了磁盘上的结构。例如，环绕信息节点 300（图 3）的虚线指出磁盘上信息节点结构在核心中的表示。核心中的数据结构是存储器块，该存储器块存储了磁盘上数据结构以及需要用来管理存储器中的数据的附加信息（但不在磁盘上）。附加信息可例如包括修改标志位 360。在信息节点（或块）中的数据如所指示的例如由写操作更新 / 修改后，使用修改标志位 460 将修改的数据标记为脏，使得信息节点（块）可随后“刷新”（存储）到磁盘。WAFL 文件系统的核心中和磁盘上格式的结构，包括信息节点和信息节点文件，在之前合并的美国专利号 5,819,292 中公开和描述，标题为 Method for MaintainingConsistent States of a File System and for Creating User-Accessible Read-Only Copies of a File System，申请人为 DavidHitz 等，于 1998 年 10 月 6 日提交。

[0054] 图 4 是可有利地用于本发明的文件缓冲树的实施例的示意性框图。缓冲树是装载到缓冲存储器 170 中的文件（例如，文件 400）的块的内部表示并且由随处可写文件系统 280 维护。根（顶级）信息节点 402，例如嵌入的信息节点，引用间接（例如级别 1）块 404。注意，取决于文件大小，可存在附加级别的间接块（例如级别 2，级别 3）。间接块（和信息节点）包含最终引用用于存储文件的实际数据的数据块 406 的指针 405。即是，文件 400 的数据包含在数据块中，且这些块的位置存储在文件的间接块中。每个级别 1 间接块 404 可包含指向多达到 1024 个数据块的指针。根据文件系统的“随处可写”特性，这些块可位于磁盘 130 上的任何位置。

[0055] 提供了文件系统布局，该文件系统布局将下层物理卷分配到存储系统，例如高速缓存文件管理器 120 的一个或多个虚拟卷 (vvol) 中。这种文件系统布局的一个例子在美国专利申请序列号 10/836,817 中描述，标题为 Extension of Write Anywhere File System Layout，申请人为 John K. Edwards 等，且转让给 Network Appliance，Inc。下层物理卷是包括高速缓存文件管理器的一个或多个磁盘组，例如 RAID 组的聚集。聚集具有其自己的物理卷块号 (pvbn) 空间并且在该 pvbn 空间内维持元数据，例如块分配结构。每个 vvol 具有

其自己的虚拟卷块号 (vvbn) 空间并且在该 vvbn 空间内维持元数据，例如块分配结构。每个 vvol 是与容器文件相关联的文件系统；容器文件是聚集中的文件，该聚集包含 vvol 使用的所有块。而且，每个 vvol 包括数据块和间接块，该间接块包含指向其它间接块或数据块的块指针。

[0056] 在一个实施例中，pvbn 用作为存储在 vvol 中的文件（例如文件 400）的缓冲树内的块指针。该“混合”vvol 实施例涉及在双亲间接块（例如信息节点或间接块）中只插入 pvbn。在逻辑卷的读取路径上，“逻辑”卷 (vol) 信息块具有一个或多个指针，该指针引用一个或多个 fsinfo 块，每个 fsinfo 块转而指向信息节点文件及其对应的信息节点缓冲树。通常，vvol 上的读取路径是一样的，跟随用于找到块的适当位置的 pvbn（而不是 vvbn）；在该上下文中，vvol 的读取路径（及对应的读取性能）基本上类似于物理卷。从 pvbn 到磁盘，dbn 的转换发生在存储操作系统 200 的文件系统 /RAID 系统的边界。

[0057] 在一个说明性双 vbn 混合（“灵活”）vvol 实施例中，pvbn 及其对应的 vvbn 插入到文件的缓冲树的双亲间接块中。即是，pvbn 和 vvbn 作为每个块的一对指针存储在大多数缓冲树结构中，该缓冲树结构具有指向其它块，例如级别 1(L1) 间接块、信息节点文件级别 0(L0) 块的指针。图 5 是可有利地用于本发明的文件 500 的缓冲树的说明性实施例的示意性框图。根（顶级）信息节点 502，例如嵌入的信息节点，引用间接（例如级别 1) 块 504。注意，取决于文件的大小可存在附加级别的间接块（例如级别 2、级别 3)。间接块（和信息节点）包含最终引用用于存储文件的实际数据的数据块 506 的 pvbn/vvbn 指针对结构 508。

[0058] pvbn 引用聚集的磁盘上的位置，而 vvbn 引用 vvol 的文件内的位置。使用 pvbn 作为间接块 504 中的块指针 508，提供了在读取路径中的效率，而使用 vvbn 块指针提供了访问所请求的元数据的效率。即是，当释放文件的块时，文件中的双亲间接块包含容易得到的 vvbn 块指针，这避免了与访问属主映射 (owner map) 以执行 pvbn 到 vvbn 转换相关联的延迟；而且，在读取路径上，pvbn 是可得到的。

[0059] 如所提到的，每个信息节点在其数据段中具有 64 个字节，取决于信息节点文件的大小（例如大于 64 字节的数据），该 64 个字节可用作为指向其它块的块指针。对于传统和混合卷，该 64 个字节实现为 16 个块指针，即 16 个 (16)4 字节的块指针。对于说明性的双 vbn 灵活卷，信息节点的 64 个字节实现为 8(8) 对 4 字节的块指针，其中每一对是 vvbn/pvbn 对。此外，传统或混合卷的每个间接块可包含达 1024 个 (pvbn) 指针；然而，双 vbn 灵活卷的每个间接块具有达 510 对 (pvbn/vvbn) 指针。

[0060] 而且，一个或多个指针 508 可包含特定 ABSENT 值以表示该一个或多个指针引用的一个或多个对象（例如间接块或数据块）没有本地存储（例如在高速缓存卷 150 上），并且因此该对象必须从原始服务器 180 的原始卷 185 获取（取回）。在说明性实施例中，文件系统 280 的 Load_Block() 函数 284 解释了每个指针的内容并且，如果所请求的块是 ABSENT，则启动使用例如 NRV 协议将对数据的适当请求（例如远程获取操作）传输到原始服务器 180。

[0061] 应当注意，高速缓存卷 150 说明性地实现为灵活 vvol，而原始卷 185 可以是灵活 vvol 或传统卷，主要是因为使用逻辑文件协议 (NRV)。如所提到的，传统卷和灵活 vvol 区别在于它们的间接块格式；然而，在网络高速缓冲系统的情况中，间接块格式的不同是不相关的。换句话说，因为在高速缓存卷和原始卷之间没有物理关系，原始卷的类型是不相关的。

[0062] 图 6 是可有利地用于本发明的聚集 600 的一个实施例的示意性框图。Lun(块)602、目录 604、qtree606 和文件 608 可包含在 vvol610 内,例如双 vbn 灵活 vvol,该 vvol 转而包含在聚集 600 内。说明性地,聚集 600 在 RAID 系统的上层,RAID 系统由至少一个 RAID plex650 表示(取决于是否镜像了存储配置),其中每个 plex650 包括至少一个 RAID 组 660。每个 RAID 组进一步包括多个磁盘 630,例如一个或多个数据 (D) 磁盘和至少一个 (P) 奇偶磁盘。

[0063] 聚集 600 类似于常规存储系统的物理卷,而 vvol 类似于该物理卷内的文件。即是,聚集 600 可包括一个或多个文件,其中每个文件包含 vvol610 且其中 vvol 所消耗的存储空间的总和在物理上小于(或等于)整个物理卷的大小。聚集使用定义了由物理卷的磁盘提供的块的存储空间的物理 pvbn 空间,而(在文件内的)每个嵌入的 vvol 使用逻辑 vvbn 空间以将那些块例如组织为文件。每个 vvbn 空间是对应于文件内位置的独立的号的集合,所述位置接着变换成磁盘上的 dbn。因为 vvol610 也是逻辑卷,因此它在其 vvbn 空间中具有自己的块分配结构(例如活动、空间和摘要映射)。

[0064] 容器文件是聚集中包含 vvol 使用的所有块的文件。容器文件是支持 vvol 的(对于聚集的)内部特征;说明性地,每个 vvol 有一个容器文件。类似于文件通道中的纯逻辑卷,容器文件是聚集中的隐藏文件(用户不可访问),该文件含有 vvol 使用的每个块。聚集包括说明性的隐藏元数据根目录,该根目录包含 vvol 的子目录:

[0065] WAFL/fsid/ 文件系统文件,存储标签文件

[0066] 具体地,对于聚集中的每个 vvol,物理文件系统 (WAFL) 目录包括子目录,子目录的名称是 vvol 的文件系统标识符 (fsid)。每个 fsid 子目录 (vvol) 包含至少两个文件,文件系统文件和存储标签文件。说明性地,存储标签文件是包含类似于存储在常规 raid 标签中的元数据的 4kB 文件。换句话说,存储标签文件是 raid 标签的模拟,并且如所述的,包含关于 vvol 的状态的信息,诸如 vvol 的名称、通用唯一标识符 (uuid) 和 vvol 的 fsid、其是否联机、被创建或被破坏等。

[0067] 图 7 是聚集 700 的磁盘上表示的示意性框图。存储操作系统 200,例如 RAID 系统 240,用包括用于聚集的“物理”volinfo 块 702 的 pvbn1 和 2 来装配 pvbn 的物理卷以创建聚集 700。volinfo 块 702 包含指向 fsinfo 块 704 的指针,每个 fsinfo 块 704 可表示聚集的一个快照。每个 fsinfo 块 704 包括指向节点信息文件 706 的块指针,所述节点信息文件 706 包含多个文件的信息节点,包括属主映射 710、活动映射 712、摘要映射 714 和空间映射 716,以及其它特定元数据文件。信息节点文件 706 进一步包括根目录 720 和“隐藏”元数据根目录 730,后者包括具有相关于 vvol 的文件的命名空间,其中用户不能“看见”文件。隐藏元数据根目录包括 WAFL/fsid/ 目录结构,该目录结构包含文件系统文件 740 和存储标签文件 790。注意,聚集中的根目录 720 是空的;相关于聚集的所有文件组织在隐藏元数据根目录 730 内。

[0068] 如果 vvol 是稀疏卷,对于每个 vvol,隐藏元数据根目录 730 还包括稀疏配置元文件(“稀疏配置文件”732)。稀疏配置文件 732 因此与稀疏卷关联并且为此(尤其)标识原始服务器 180 的主机名和原始卷 185。在稀疏卷的安装过程中,获取稀疏配置文件 732 且将其转换为核心中的格式。要注意的是,稀疏配置文件还包括指明稀疏卷是否是高速缓存卷 150 的标识符。这些标识符允许高速缓存文件管理器 120 确定它应当对于不同的客户端

请求执行远程更新还是本地更新；如这里进一步描述的，网络高速缓存系统环境 100 的高速缓存文件管理器说明性地执行远程更新。

[0069] 除了实现为具有组织为容器映射的等级 1 的块的容器文件，文件系统文件 740 包括块指针，该块指针引用实现为 vvol750 的不同文件系统。聚集 700 在专门保留的信息节点号处维护这些 vvol750。每个 vvol750 还在其 vvol 空间内具有专门保留的信息节点号，该信息节点号尤其用于块分配位图结构。如所记录的，块分配位图结构，例如活动映射 762、摘要映射 764 和空间映射 766，位于每个 vvol 中。

[0070] 具体地，每个 vvol750 具有和聚集一样的信息节点文件结构 / 内容，不同的只是没有属主映射且在隐藏元数据根目录 780 中没有 WAFL/fsid/ 文件系统文件，存储标签文件目录结构。为此，每个 vvol750 具有指向一个或多个 fsinfo 块 800 的 volinfo 块 752，每个 fsinfo 块 800 可与 vvol 的活动文件系统一起表示一个快照。每个 fsinfo 块转而指向一个信息节点文件 760，如所提到的，该信息节点文件 760 除了上面所述的不同点外，和聚集有相同的信息节点结构 / 内容。每个 vvol750 具有其自己的信息节点文件 760 和具有对应的信息节点号的不同的信息节点空间，以及其自己的根 (fsid) 目录 770 和可从其它 vvol 单独输出的文件的子目录。

[0071] 包含在聚集的隐藏元数据根目录 730 内的存储标签文件 790 是功能类似于常规 raid 标签的小文件。raid 标签包括关于存储系统的物理信息，例如卷名称；该信息装载到存储标签文件 790 中。说明性地，存储标签文件 790 包括相关联的 vvol750 的名称 792、vvol 的联机 / 脱机状态 794、及相关联的 vvol 的其它身份和状态信息 796（是否其处在创建或破坏的过程中）。

[0072] D. 稀疏的卷

[0073] 如所记录的，高速缓存卷 150 说明性地实现为稀疏的卷，并且由此术语“高速缓存卷 150”和“稀疏卷 150”在下文中可互换使用。由卷 (vvol) 的磁盘上结构的特定标记来标识稀疏卷 150，以表示包括具有缺少块的文件。图 8 是磁盘上结构的示意性框图，其说明性地为示例 fsinfo 块 800。fsinfo 块 800 包括持久一致性点像 (PCPI) 指针 805 的集合、稀疏卷标志字段 810、信息节点文件的信息节点 815 和在替换的实施例中的附加字段 820。PCPI 指针 805 是指向与文件系统关联的 PCPI（快照）的双 vbn(vvbn/pvbn) 成对指针。稀疏卷标志字段 810 标识由 fsinfo 块描述的 vvol 是否为稀疏的。在说明性的实施例中，在字段 810 中插入标志以标识卷是稀疏的。稀疏卷标志字段 810 可进一步实现为用于标识与 fsinfo 块关联的 vvol 的类型的类型字段。信息节点文件的信息节点 815 包括包含根级指针的信息节点，该根级指针指向与 fsinfo 块关联的文件系统的信息节点文件 760（图 7）。

[0074] 用专门的 ABSENT 值将文件的适当块指针做标记（标志）以指明，稀疏卷 150 内的某些块，包括数据和 / 或间接块没有物理地位于提供卷的高速缓存文件管理器上。专门的 ABSENT 值进一步告知文件系统，数据将从可替换源，即原始服务器 180 获得。响应于数据访问请求，文件系统 280 的 Load_Block() 函数 284 检测文件的适当块指针是否标记为 ABSENT，并且如果是，将远程 NRV 获取（例如读取）操作信息从高速缓存文件管理器传输到原始服务器以获取所请求的数据。说明性地，获取操作请求在原始卷 185 上存储的文件的一个或多个文件块号 (fbn)。应当注意，虽然根据单个原始卷来撰写本发明书，本发明的原理可用于以下环境：其中由多个原始卷支持单个稀疏卷，每个原始卷可支持稀疏卷的全部

或其子集。如所述的，本教导不应当限于单个原始卷。

[0075] 原始服务器 180 从其存储设备获取所请求的数据并且将所请求的数据返回到高速缓存文件管理器 120，该高速缓存文件管理器 120 处理数据访问请求并且将返回的数据存储在其存储器 124 中。随后，文件系统 280 在写分配过程中将存储在其存储器中的数据“刷新”（写入）到本地磁盘。这可以响应于数据被标记为“脏”或向文件系统表示数据必须被写分配的其它符号。根据过程的说明性的随处可写策略，文件系统 280 将指针值（不是 ABSENT 值）分配到文件的间接块，以由此标识本地存储在高速缓存卷 150 内的数据位置。因此，不再需要远程获取操作来访问数据。

[0076] 应当注意，在网络 140 上高速缓存文件管理器 120 和原始服务器 180 之间传输的所有 NRV 消息涉及相对于物理磁盘地址的逻辑文件地址。由此，不需要相对于原始服务器存储来制定高速缓存文件管理器存储的大小。当将所请求的数据提供到高速缓存文件管理器时，该数据被写分配并且依从适当的 vvbn（和 / 或 pvbn）块编号。换句话说，高速缓存卷 150 的写分配完全不同于原始卷 185 上的写分配。

[0077] 可有利地用于本发明的写分配过程的一个例子在美国专利申请序列号 10/836,090 中描述，标题为 Extension of Write AnywhereFile Layout Write Allocation，申请人为 John K. Edwards，该申请因此被合并以作为参考。概括地叙述，当写分配 vvol 内的块时，块分配在灵活 vvol 和聚集上并行地进行，并且写分配器 282（图 2）在聚集中选择实际的 pvbn 且在 vvol 中选择 vvbn。写分配器调整聚集的块分配位图结构，例如活动映射和空间映射，以记录选择的 pvbn，并且调整 vvol 的类似结构，以记录所选择的 vvbn。vvol 的 vvid（vvol 标识符）和 vvbn 在由所选择的 pvbn 定义的条目处被插入到聚集的属主映射 710 中。所选择的 pvbn 还被插入到目标 vvol 的容器映射（未示出）中。最后，用指向所分配块的一个或多个块指针来更新所分配块的间接块或信息节点文件双亲。更新操作的内容依赖于 vvol 实施例。对于双 vbn 混合 vvol 实施例，pvbn 和 vvbn 都插入到间接块或作为块指针的信息节点中。

[0078] E. 网络高速缓存系统操作

[0079] 本发明涉及网络高速缓存系统 100，其具有连接到原始服务器 180 的多协议高速缓存文件管理器 120，以响应于计算机网络 140 上多协议客户端 110 发出的数据访问请求而提供由文件管理器服务的数据的存储虚拟化。多协议高速缓存文件管理器 120 包括配置为管理稀疏卷的文件系统 280，该文件系统 280“虚拟化”数据的存储空间，以由此提供高速缓存功能而使得多协议客户端可以访问数据。为此，高速缓存文件管理器还包括存储操作系统 200 的多协议引擎，该引擎配置为将多协议客户端数据访问请求变换成由高速缓存文件管理器和原始服务器 180 都可执行的通用文件系统原始操作。

[0080] 图 9 是根据本发明的实施例示出用于处理数据修改访问请求的过程 900 的步骤的流程图。如这里使用的，数据修改访问请求涉及修改高速缓存文件管理器 120 的高速缓存卷 150 的任何操作。这种修改操作的例子包括创建（文件）、设置属性和写入操作。过程 900 开始于步骤 902 且进行到步骤 904，其中在高速缓存文件管理器 120 处接收客户端写入请求。在步骤 906，多协议引擎的适当协议层将写入请求转换为通用文件系统写入消息以传送到文件系统 280。

[0081] 在步骤 908，文件系统确定文件系统写入消息是否指向于高速缓存卷 150，即配置

为支持远程更新操作的稀疏卷。说明性地，文件系统通过检查 fsinfo 块 800 和稀疏配置文件 732 来做出该确定。如所提到的，fsinfo 块 800 具有稀疏卷标志 810，如果声明了该稀疏卷标志 810，则标识卷为稀疏卷。此外，稀疏配置文件 732 包含在应用类型中标识稀疏卷 150 的标识符，即标识稀疏卷 150 是否支持数据修改访问请求的远程更新。如果写入消息不指向高速缓存卷，文件系统将文件系统写入消息传递到文件的常规写入处理器，以作为原始写入操作请求进行处理（步骤 910）并且该过程在步骤 922 结束。

[0082] 然而，如果写入消息指向高速缓存卷 150，在步骤 912 文件系统将写入消息转发到 RUE292。在步骤 914，RUE292 将通用文件系统写入消息转换为远程更新请求，并且在步骤 916，将更新请求发送到泵模块 298。在说明性的实施例中，泵模块的泵工作者线程接收请求，接着为该请求在其它请求中安排优先次序。在步骤 918，将远程更新请求变换为 NRV 写入消息，并且在步骤 920，在网络 140 上将 NRV 写入消息发送到原始服务器 180，以由服务器上的文件系统执行。接着过程在步骤 922 结束。

[0083] 图 10 是根据本发明的实施例的示出用于处理非数据修改访问请求的过程 1000 的步骤的流程图。如这里使用的，非数据修改访问请求涉及不修改高速缓存文件管理器 120 的高速缓存卷 150 的任何操作。非修改操作的一个例子是读取操作。过程 1000 在步骤 1002 开始，并且进行到步骤 1004，其中在高速缓存文件管理器 120 接收客户端读取请求。在步骤 1006，多协议引擎的适当协议层将读取请求转换为通用文件系统读取消息，以传送到文件系统 280，在步骤 1008，该文件系统 280 将消息传递到文件系统的常规读取处理器，以作为原始读取操作请求来处理。

[0084] 在步骤 1010，确定所请求的数据是否驻留在高速缓存文件管理器的本地高速缓存上。说明性地，文件系统通过使用例如 Load_Block()284 函数装载一个或多个块且检查每个块的块指针以确定该块指针是否标记为 ABSENT 来做出所述确定。如果块不是缺少的，即所请求的数据驻留在本地高速缓存上，在步骤 1012 文件系统 280 服务读取消息 / 请求（如前所述）并且过程在步骤 1032 结束。

[0085] 然而，如果块是缺少的，即所请求的数据未驻留在本地高速缓存上，在步骤 1014 文件系统将读取消息转换为发送到泵模块 298 的获取请求。泵模块的泵工作者线程接收请求，接着为该请求在其它请求中安排优先次序。在步骤 1016，泵线程维持用于存储获取请求的占位符，直到接收了响应。在步骤 1018，泵线程和 NRV 模块 295 协同以将获取请求变换为 NRV 读取消息，并且在步骤 1020，在网络 140 上将 NRV 读取消息发送到原始服务器 180 以由服务器执行。

[0086] 在步骤 1022，原始服务器用获取的数据响应于高速缓存文件管理器（泵线程），并且在步骤 1024，泵线程和文件系统的填充处理器协同以通过例如使用获取的数据执行填充操作来服务在泵模块维持着占位符的挂起的读取 / 获取请求。在步骤 1026，文件系统用所请求的数据构造回复，并且在步骤 1028，将该回复返回到客户端。随后在步骤 1030，在文件系统执行写分配以将获取的数据存储在高速缓存文件管理器的一个或多个本地存储设备上并且过程在步骤 1032 结束。

[0087] F. 高速缓存一致性

[0088] 在本发明的通常网络高速缓存系统实施例中，多个客户端 110 可耦合到多个高速缓存文件管理器 120 的每一个，并且客户端和文件管理器都可耦合到原始服务器 180。因此

可能的是，在该通常系统实施例中，原始卷 185 可由客户端和 / 或高速缓存文件管理器 120 修改。所以，需要高速缓存一致性策略以确保由客户端直接从原始服务器 180 或经由高速缓存文件管理器 120 访问的数据总是一致的。根据本发明，用在网络高速缓存系统 100 中的高速缓存一致性策略规定了将高速缓存文件管理器 120 配置为与原始服务器 180 相符合，以便在将数据递送到客户端 110 之前确定该数据是否发生了变化。

[0089] 响应于指向例如文件的特别存储对象的客户端数据访问请求，例如读取请求，高速缓存文件管理器 120 的文件系统 280 将按需获取 (FOD) 的请求发送到原始服务器 180，以请求文件属性，诸如修改时间、链接数目、创建时间等的最近副本。任何属性中的变化指明自从文件最后一次高速缓存在文件管理器中后，该文件被修改过。由此，高速缓存文件管理器触发了在其本地高速缓存上存储的当前文件的弹出。高速缓存文件管理器接着使用 NRV 读取消息生成适当的获取操作，以从原始服务器获取所请求的数据。

[0090] 图 11 是根据本发明的实施例的示出实现高速缓存一致性策略的过程 1100 的步骤的流程图。过程 1100 在步骤 1102 开始并且进行到步骤 1104，其中在高速缓存文件管理器接收客户端数据访问请求，例如读取请求。在步骤 1106，将请求转换为文件系统读取消息，以传送到文件系统 280，在步骤 1108，文件系统 280 将消息传递到文件系统的常规读取处理器，以作为原始读取操作请求处理。在步骤 1110，读取处理器例如使用 Load_Inode()288 函数获取在读取请求 / 消息中涉及的文件的信息节点。

[0091] 在步骤 1112，文件系统还将读取消息作为 FOD 请求传递到泵模块，以从原始服务器获取信息节点的属性。在步骤 1114，泵线程维持用于存储 FOD 请求的占位符，直到接收了响应。在步骤 1116，泵模块和 NRV 模块协同以将 FOD 请求变换为 NRV 读取消息，并且在步骤 1118，在网络 140 上将 NRV 读取消息发送到原始服务器 180，以由服务器执行。在步骤 1120，原始服务器用属性响应高速缓存文件管理器（泵线程），并且在步骤 1122，泵线程和文件系统的填充处理器协同以通过例如使用响应执行填充操作来服务在泵模块维持着占位符的挂起读取 /FOD 请求。

[0092] 应当注意，不具有数据（即零长度读取或“验证”）的填充操作仅携带属性；因此，在步骤 1124，填充处理器确定（从原始服务器接收的）所请求文件的属性和当前存储在高速缓存文件管理器上的该文件的属性是否不同。对于后者，例如通过检查存储在文件的信息节点 300 中的访问和 / 或修改时间戳 316 来确定存储在高速缓存文件管理器上的文件的属性状态。注意，NRV 读取消息的特性是在 NRV 响应中返回的任何数据还包括文件的最近属性。零长度读取（验证）因此等效于不获取任何数据而取回文件的最近属性。

[0093] 如果在属性中没有区别（属性没有变化），填充处理器触发了信息节点（文件）属性已经被验证 (1126) 的确认。因此在高速缓存文件管理器 120 和原始服务器 180 之间的 NRV 交互实质上是“没有操作”，这在系统中引入了额外的延迟（至少在最简单的高速缓存一致性策略中）。在步骤 1128，文件系统搜索本地高速缓存以确定客户端请求的数据是否存在于高速缓存文件管理器上。如果是，在步骤 1130 文件系统服务读取请求 / 消息（如前所述）并且过程在步骤 1136 结束。

[0094] 然而，如果所请求的数据（或其部分）未驻留在本地高速缓存上（即数据遗失），在步骤 1132 文件系统将读取消息转换为获取请求，该获取请求最终被发送到原始服务器 180 以获得缺少的数据（如前所述）。注意，来自原始服务器的响应包括遗失数据和文件的

最后属性。还要注意，如果在属性中存在不同（如在步骤 1124 确定的），过程继续到步骤 1132。在步骤 1134，确定是否那些属性发生了变化（即在高速缓存文件管理器上初次验证和获取了遗失数据的时间之间属性是否发生了变化）。如果是，过程返回到步骤 1132。否则，过程继续到步骤 1130。

[0095] 要注意，在确定客户端请求的数据是否存在于高速缓存文件管理器 120 上之前验证所述数据。这是因为，如果数据存在于高速缓存文件管理器上，即使在验证和服务数据之间在原始服务器 180 上发生对该数据的更新，也不向客户端发出。在该后一种情况下，那些操作被认为是“覆盖操作”，并且通过将读取请求作为首次发生来对待而将那些操作串行化。还要注意，考虑网络高速缓存系统部署，其中多个客户端访问多个高速缓存文件管理器和 / 或原始服务器，则属性可以变化。

[0096] 在说明性的实施例中，在网络高速缓存系统部署上没有显式的锁定。然而，网络高速缓存系统依赖的语义是，与写入操作覆盖的读取操作（即在验证之前不发生写入）可在写入之前返回读取。换句话说，验证响应指明对于文件没有属性发生变化并且该文件数据可从高速缓存文件管理器的高速缓存卷 150 提供（如果可能）。当随后服务来自高速缓存卷的该数据时，高速缓存文件管理器 120 像读取操作发生在写入操作之前一样地来操作。

[0097] 清楚地，在高速缓存命中的情况下，网络高速缓存系统 100 保持语义。在部分高速缓存未中的情况下，网络高速缓存系统 100 通过有效地从擦除 (scratch) 开始来保持语义。对于后者，假设客户端发送 32kB 的读取请求并且高速缓存文件管理器仅遗失了该请求的 4kB 的块（遗失数据不在高速缓存卷上）。该情况的正常响应是，为高速缓存文件管理器发送 4kB 的 NRV 读取消息，以填充该遗失数据，并伴有隐式的验证（因为每个读取返回文件属性）。还假设之前的显式验证指明数据没有什么变化，但在显式验证和发送 4kB 的 NRV 读取消息之间发生了介入的写操作。因为在读取响应中属性发生了变化（由伴随有响应的隐式验证来表示），高速缓存文件管理器检测到该介入的写入。这转而使得高速缓存文件管理器 120 在其高速缓存卷 150 上弹出其文件的副本并且使用 NRV 读取消息来生成适当的获取操作，以从原始服务器 180 获取所请求的数据。该情况表示了写入操作可引起额外的和浪费的读取操作。

[0098] 根据本发明的一个方面，泵模块 298 可用于缓解这种不足。泵模块实现流控制并且新的网络高速缓存系统体系结构提供了另一种形式的流控制，该流控制实质上代理了向原始服务器 180 的读取操作，而不用经由普通文件系统读取处理器为它们服务。即是，响应于为服务客户端请求而难以将文件的数据装载到其本地高速缓存中，高速缓存文件管理器 120 切换到这样的模式 :将指向该文件的读取操作传递到 RUE292（类似于写操作）且传递到原始服务器 180 上，而是通过文件系统 280 传递到读取处理器。原始服务器接着使用标准的流控制和原子性机制，以便将单个响应返回到该读取操作。

[0099] G. 优先化

[0100] 根据本发明的一个方面，由高速缓存文件管理器执行提前读取操作，并且因此当在客户端请求和推测的提前读取请求之间存在差异时，文件管理器实现了优先化。该特征对于网络高速缓存系统实现的优点在于，因为它不会“看见”所有的客户端请求，因此通常当做出提前读取的决定时，原始服务器不必具有和高速缓存文件管理器一样多的知识。因为高速缓存文件管理器具有在其上执行的多协议引擎，它可做出与通常由原始服务器做出

的相同的提前读取的决定，即使在文件管理器和服务器之间存在高速缓存卷。特别地，高速缓存文件管理器使用和由原始服务器使用的相同的提前读取引擎，并且因此像原始服务器一样生成相同的提前读取请求。为了请求的优先化，网络高速缓存系统实现将请求作为两个不同的优先级范围对待，其中客户端请求的优先级在推测的提前读取之上，并且如果系统是饱和的，则丢弃推测的提前读取请求。

[0101] H. 高速缓存弹出

[0102] 如所记录的，截取器 294 编码高速缓存弹出策略，以便当本地高速缓存（例如高速缓存卷 150）变满时回收存储空间。在高速缓存文件管理器 120 的高速缓存卷 150 小于在原始服务器 180 的原始卷 185 上存储的工作区的情况下，频繁地出现高速缓存弹出决定。当接收客户端请求时，高速缓存文件管理器需要释放卷存储空间以高速缓存（存储）那些请求。在释放空间时，必须从高速缓存卷 150 收回一些数据。在说明性的实施例中，截取器 294 实现为扫描器，其配置为当需要空间时，(i) 在高速缓存卷 150 内“行走”以扫描存储在卷上的文件的缓冲树和 (ii) 做出关于应当收回哪些之前被高速缓存的数据的决定。

[0103] 说明性地，高速缓存弹出策略是贯穿信息节点文件的循环处理，优点是不需要维持全局的最近最少使用 (LRU) 的列表。为此，截取器 294 以循环法 (round robin) 方式扫描信息节点文件，例如在信息节点的起点开始，前进到终点并且接着在该文件的起点重新开始，且任意地收回其经过的每个完整文件（直到满足了需要的空闲空间）。因此，当需要空间时策略随机地收回文件，但具有特性：在信息节点文件被完全地遍历之前，相同的文件不会被收回两次。如果高速缓存卷是繁忙的，则非常可能的是在任何给定的时间将工作区的大部分高速缓存。然而，如果错误地收回了“流行”的文件，在截取器遍历了整个信息节点文件之前策略将不会再次收回该文件。

[0104] 图 12 是根据本发明的实施例的示出实现高速缓存弹出策略的过程 1200 的步骤的流程图。过程在步骤 1202 开始，并且进行到步骤 1204，其中截取器初始化到信息节点文件的第一信息节点。在步骤 1206，响应于例如高速缓存卷变为完全填充，截取器被唤醒（调用）。在步骤 1208，截取器“收回”第一信息节点并且在步骤 1210，继续收回随后的信息节点（文件），直到在卷上有足够的可用存储空间。实质上，仅当需要收回高速缓存卷上的存储空间时，截取器被激活且扫过信息节点文件。说明性地，通过传递缓冲树到删除现有块的“僵 (zombie)”系统来收回信息节点或文件（或更具体地，文件的信息节点缓冲树）并且该信息节点接着被在顶级具有“孔”的信息节点代替。关于这点，孔定义为高速缓存卷上的信息节点文件的未分配的段（与缺少块相反，缺少块是被分配的）。过程接着在步骤 1212 结束。

[0105] 对于高速缓存弹出策略的优化是收回（删除）信息节点的整个块，诸如删除信息节点文件块中的每个信息节点、释放信息节点文件块和将孔插入在信息节点文件中其位置处（分配新的空白信息节点文件块）。在高速缓存卷上的孔（或信息节点文件的未分配段）依从于原始服务器上的可能实际使信息节点被分配的信息节点文件块。在该后一种情况下，当客户端请求访问特别的文件时，高速缓存文件管理器仅分配信息节点文件块；在分配信息节点文件块时，高速缓存文件管理器启动获取以获得文件内容。该特定于高速缓存的格式使得能够使用文件系统默认策略，以在信息节点文件中用新的未分配信息节点来填充孔。

[0106] 在说明性的实施例中，存在两个用于激活截取器 294 的触发器。一个触发器在填充时间（其中术语“填充”表示当在高速缓存文件管理器从原始接收响应时采取的动作）发生。在填充时间，期望将任何返回的数据插入到其文件的缓冲树中；但如果没有足够的物理磁盘空间容纳该数据，通过文件系统中的空间统计来触发截取器。说明性地，检查聚集中的空闲块的数目并且基于低 - 高水线标志（例如 85％ -95％），确定触发截取器是适当的。

[0107] 截取器的另一个触发器是在文件系统一致性点 (CP) 时刻。因为高速缓存卷是灵活的 vvol，它们可和传统卷共存于相同的聚集上。当传统（或虚拟）卷扩展而消耗更多的磁盘空间时，在高速缓存卷上触发截取以限制其磁盘空间的消耗。在 CP 时间（例如每 10 秒或 CP 频繁发生）测试磁盘空间的量（聚集中的自由物理空间）。这里，写分配器 282 发信号通知截取器 294 重新启动并且释放聚集的存储空间，直到可用空间落入低于设立的低水线标志。

[0108] I. 结论

[0109] 有利地，本发明虚拟化多协议高速缓存文件管理器的存储空间，使得能够让客户端对网络高速缓存系统服务的数据快且有效地访问。不像之前的高速缓存系统，该系统需要显式的文件的句柄到对象的存储转换，新的多协议高速缓存文件管理器使得客户端能够通过使用文件系统且特别地使用由文件系统组织的存储对象（文件）的实际名称而有效地访问由网络高速缓存系统服务的数据。而且，文件系统与高速缓存文件管理器的稀疏的卷协同，以便提供以对多协议客户端透明的方式对所提供数据的存储空间虚拟化。

[0110] 尽管已经示出和描述了网络高速缓存系统的说明性实施例，所述网络高速缓存系统具有耦合到原始服务器的多协议高速缓存文件管理器，以响应于计算机网络上多协议客户端发出的数据访问请求而提供由文件管理器服务的数据的存储虚拟化，应当理解，可在本发明的精神和范围内做出各种其它适应和修改。例如，在本发明的可替换实施例中，需求生成器 296 可用于系统地获取没有本地存储在磁盘上的数据块，以用于预填充高速缓存卷。注意，在高速缓存部署中使用比原始卷 185 小很多的高速缓存卷 150（例如以提供相对于完全复制的优点）是常见的。所以，更小的高速缓存卷的预填充需要专门的需求生成器，该需求生成器配置为提供关于应当驻留在本地高速缓存的数据的智能决定，因为不是所有的原始数据都适合高速缓存。

[0111] 前面的说明已经针对了本发明的特定实施例。然而明显的是，可对描述的实施例做出其它变化和修改，而具有所述实施例的一些或全部优点。例如，明显可预期的是，本发明的教导可实现为软件，包括具有在计算机上执行的程序指令的计算机可读介质、硬件、固件或其组合。因此，仅是通过举例的方式给出了本说明书并且其并不限制本发明的范围。因此，所附权利要求的目标是覆盖落入本发明的真实精神和范围的所有这种变化和修改。

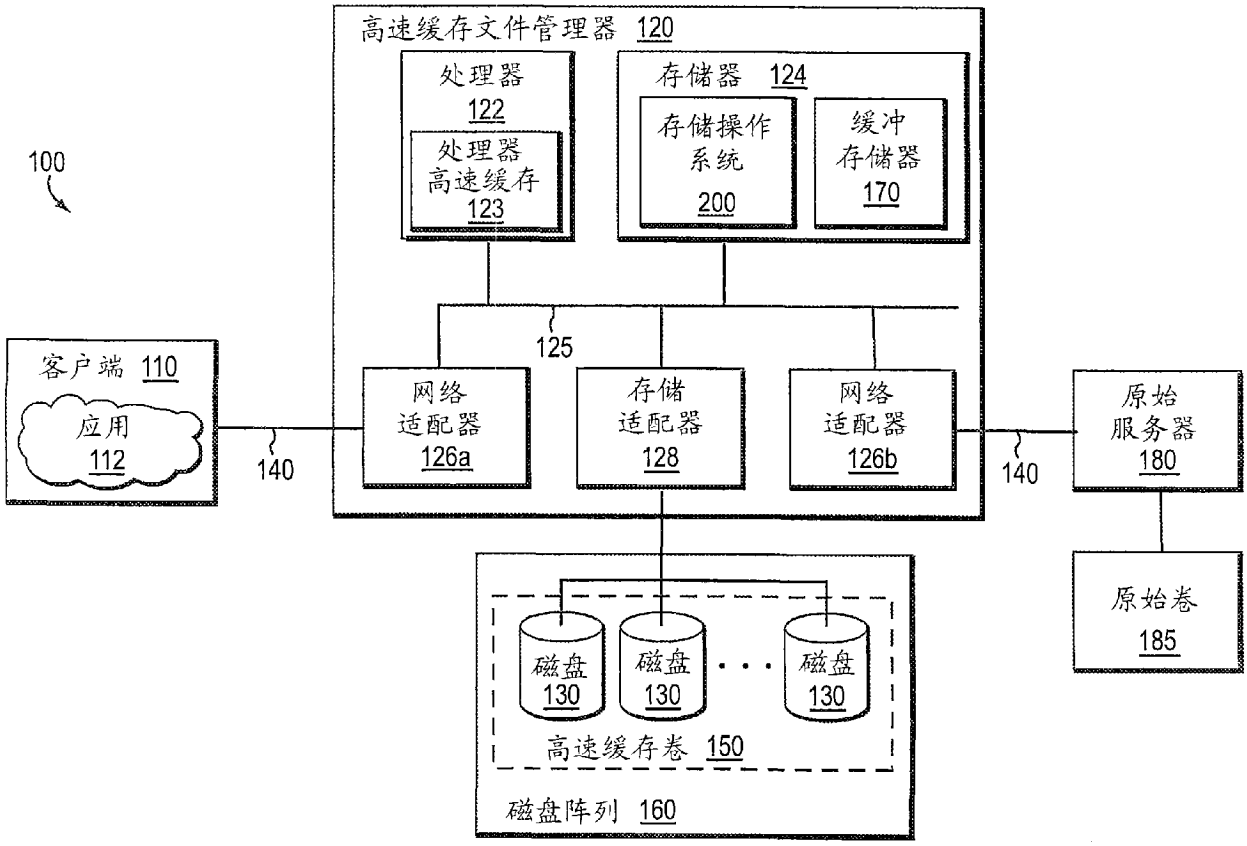

高速缓存文件管理器 120
处理器 122
处理器高速缓存 123
存储器 124
存储操作系统 200
缓冲存储器 170
100
125
客户端 110
应用 112
140
网络适配器 126a
存储适配器 128
网络适配器 126b
140
原始服务器 180
原始卷 185
磁盘 130
磁盘 130
磁盘 130
高速缓存卷 150
磁盘阵列 160

图 1

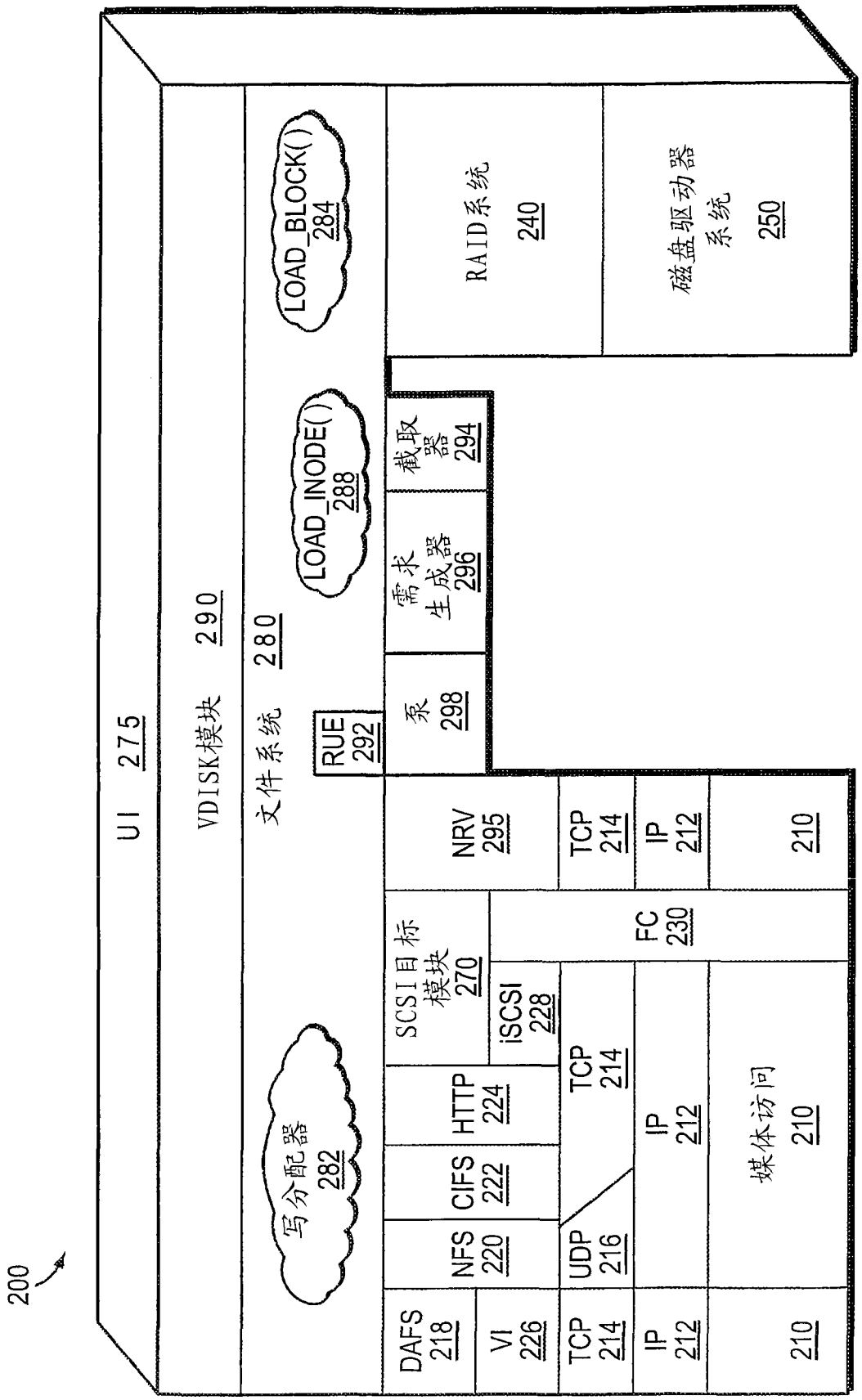

200
UI 275
VDISK模块 290
文件系统 280
写分配器 282
LOAD_BLOCK() 284
LOAD_INODE() 288
RUE 292
泵 298
需求生成器 296
截取器 294
RAID系统 240
磁盘驱动器系统 250
DAFS 218
VI 226
NFS 220
CIFS 222
HTTP 224
SCSI目标模块 270
iSCSI 228
NRV 295
TCP 214
IP 212
FC 230
媒体访问 210
UDP 216

图 2

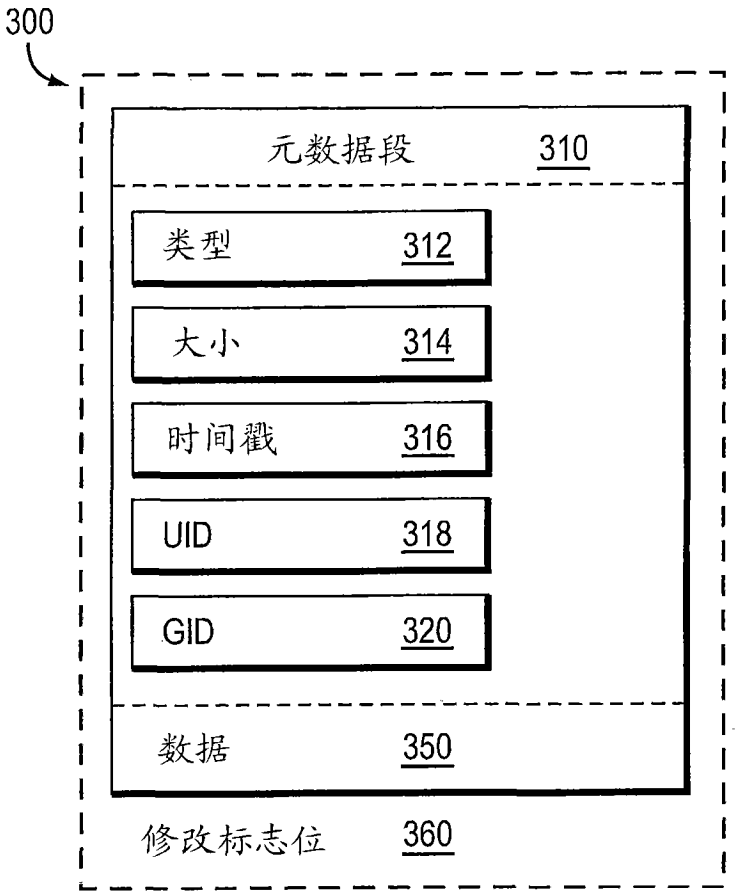

300
元数据段 310
类型 312
大小 314
时间戳 316
UID 318
GID 320
数据 350
修改标志位 360

图 3

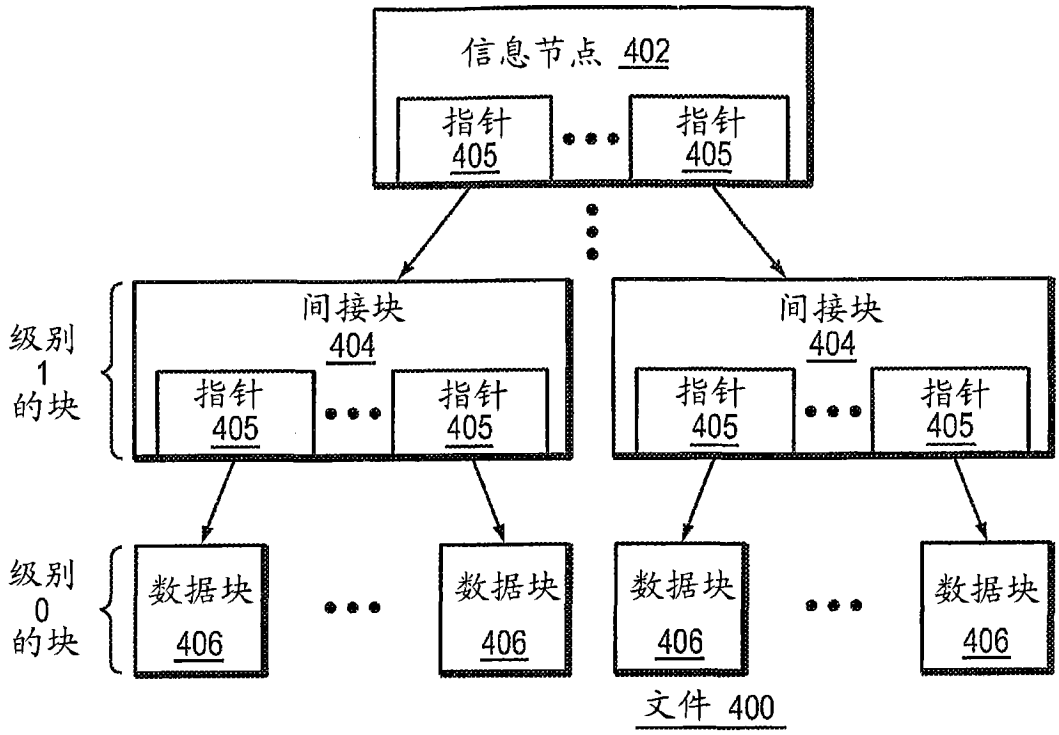

信息节点 402
指针 405
指针 405
间接块 404
间接块 404
级别 1 的块
指针 405
指针 405
指针 405
指针 405
级别 0 的块
数据块 406
数据块 406
数据块 406
数据块 406
文件 400

图 4

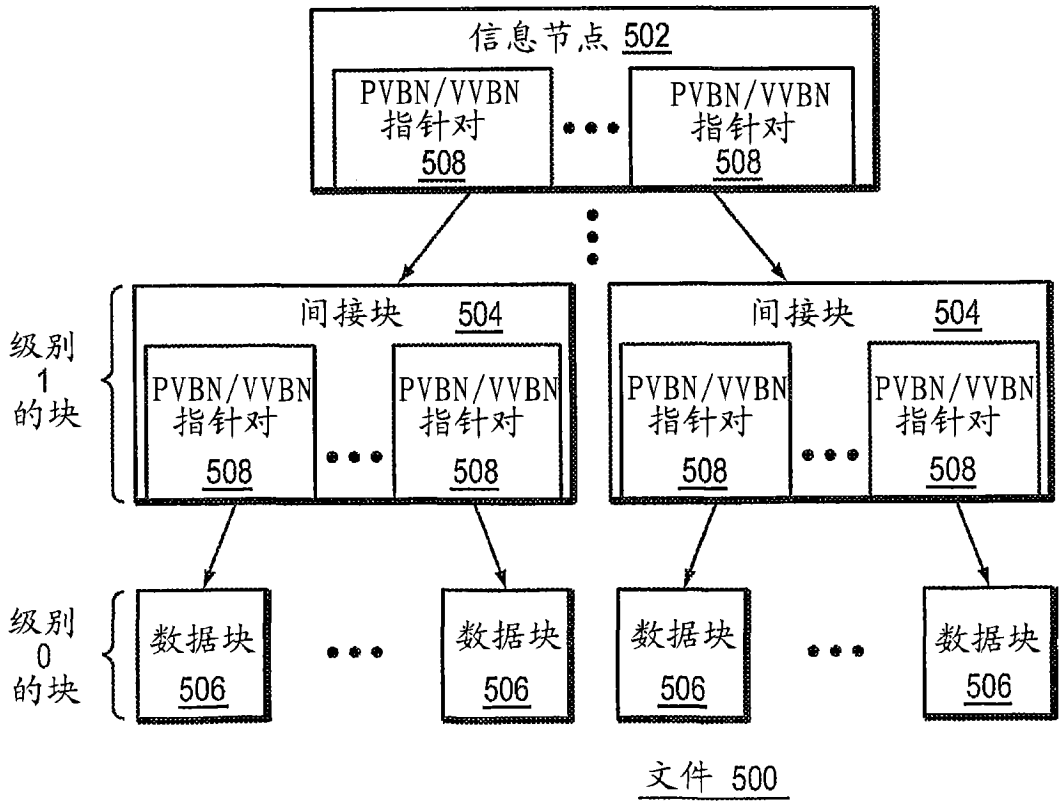

信息节点 502
PVBN/VVBN 指针对 508
PVBN/VVBN 指针对 508
间接块 504
间接块 504
级别 1 的块
PVBN/VVBN 指针对 508
PVBN/VVBN 指针对 508
PVBN/VVBN 指针对 508
PVBN/VVBN 指针对 508
级别 0 的块
数据块 506
数据块 506
数据块 506
数据块 506
文件 500

图 5

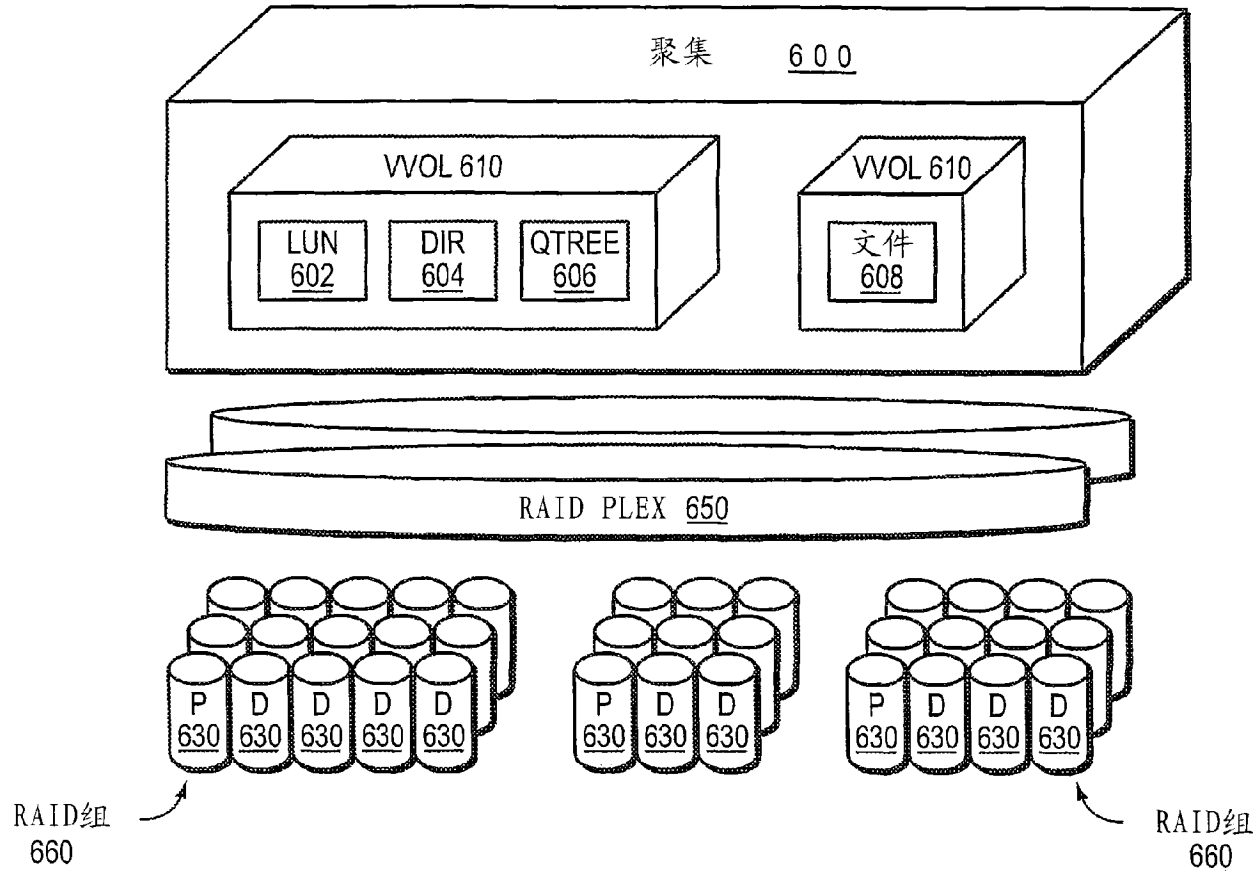

聚集 600
VVOL 610
LUN 602
DIR 604
QTREE 606
VVOL 610
文件 608
RAID PLEX 650
P 630
D 630
D 630
D 630
D 630
P 630
D 630
D 630
P 630
D 630
D 630
D 630
RAID组 660
RAID组 660

图 6

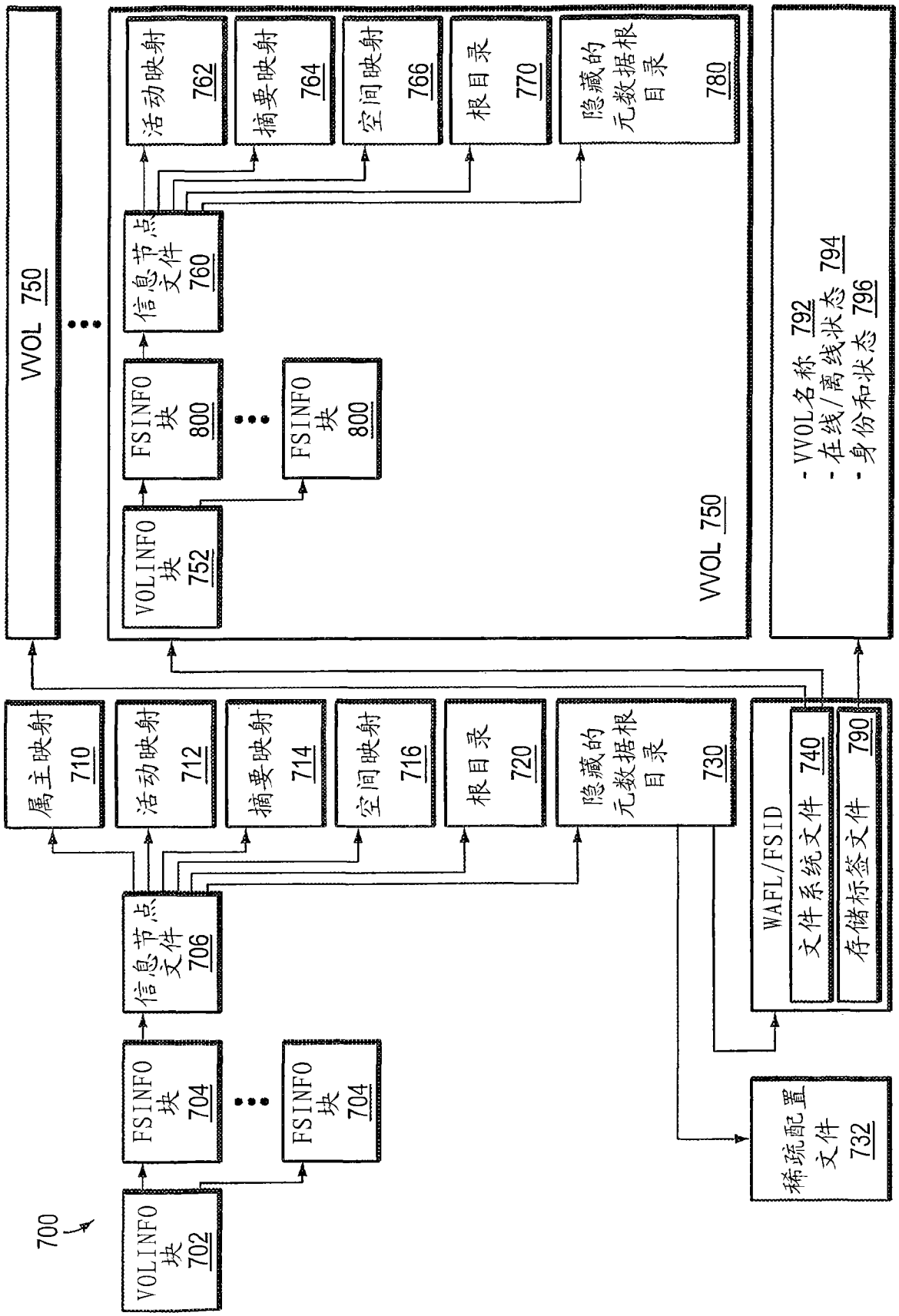

700
VOLINFO 块 702
FSINFO 块 704
FSINFO 块 704
信息节点文件 706
属主映射 710
活动映射 712
摘要映射 714
空间映射 716
根目录 720
隐藏的元数据根目录 730
稀疏配置文件 732
WAFL/FSID
文件系统文件 740
存储标签文件 790
VVOL 750
VOLINFO 块 752
FSINFO 块 800
FSINFO 块 800
信息节点文件 760
活动映射 762
摘要映射 764
空间映射 766
根目录 770
隐藏的元数据根目录 780
VVOL 750
- VVOL名称 792
- 在线/离线状态 794
- 身份和状态 796

图 7

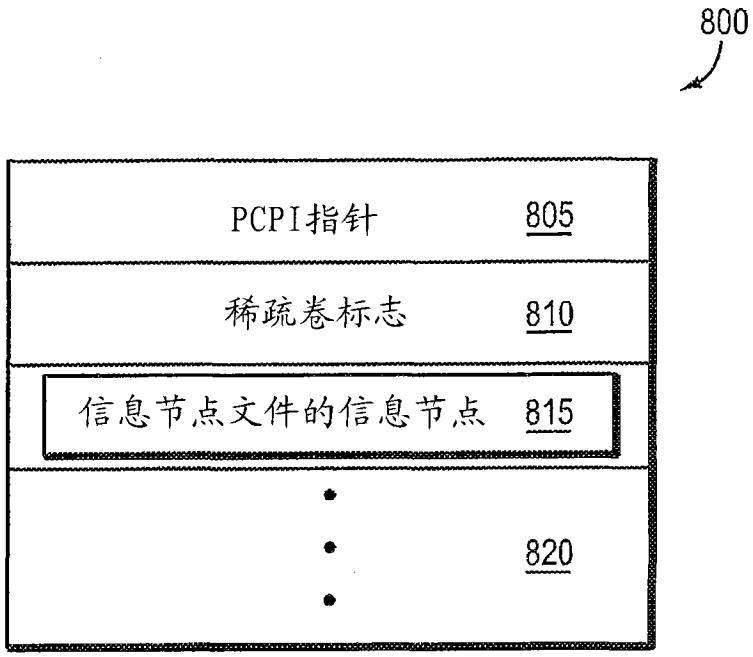


图 8

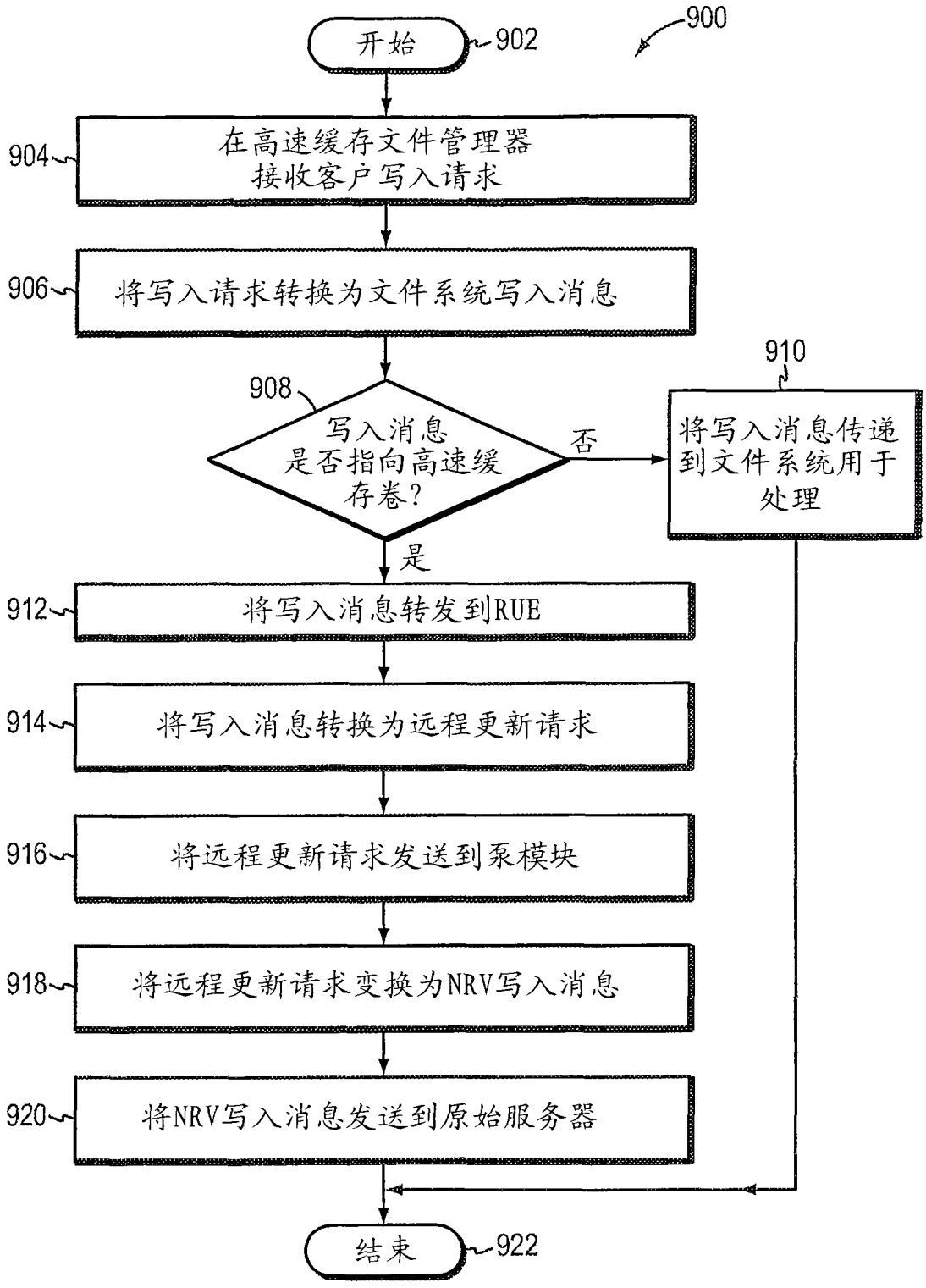

900
开始
902
在高速缓存文件管理器
接收客户写入请求
904
将写入请求转换为文件系统写入消息
906
写入消息
是否指向高速缓
存卷?
908
否
将写入消息传递
到文件系统用于
处理
910
是
将写入消息转发到RUE
912
将写入消息转换为远程更新请求
914
将远程更新请求发送到泵模块
916
将远程更新请求变换为NRV写入消息
918
将NRV写入消息发送到原始服务器
920
结束
922

图 9

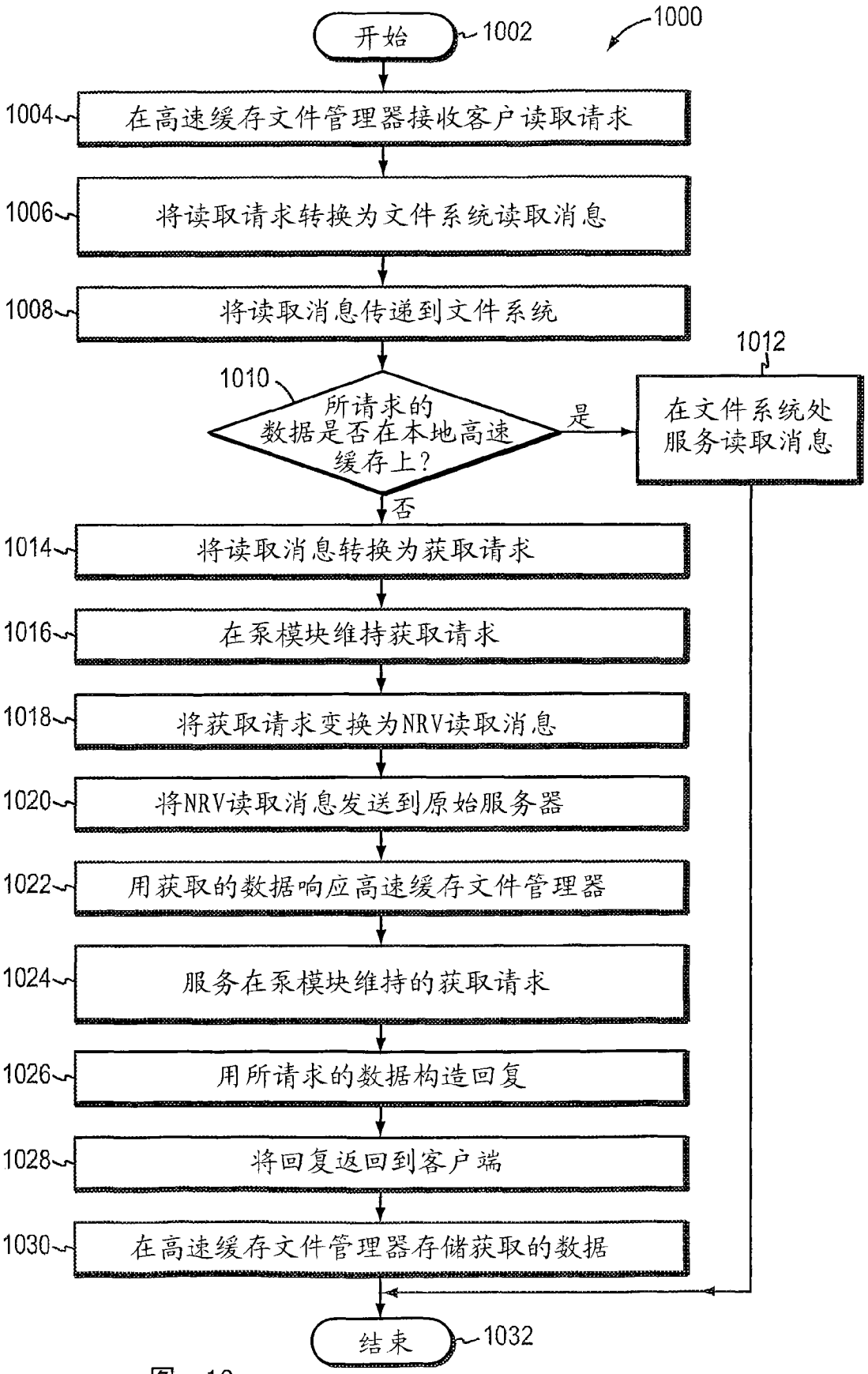

1000
开始
1002
1004
在高速缓存文件管理器接收客户读取请求
1006
将读取请求转换为文件系统读取消息
1008
将读取消息传递到文件系统
1010
所请求的数据是否在本地高速缓存上?
是
1012
在文件系统处服务读取消息
否
1014
将读取消息转换为获取请求
1016
在泵模块维持获取请求
1018
将获取请求变换为NRV读取消息
1020
将NRV读取消息发送到原始服务器
1022
用获取的数据响应高速缓存文件管理器
1024
服务在泵模块维持的获取请求
1026
用所请求的数据构造回复
1028
将回复返回到客户端
1030
在高速缓存文件管理器存储获取的数据
结束
1032

图 10

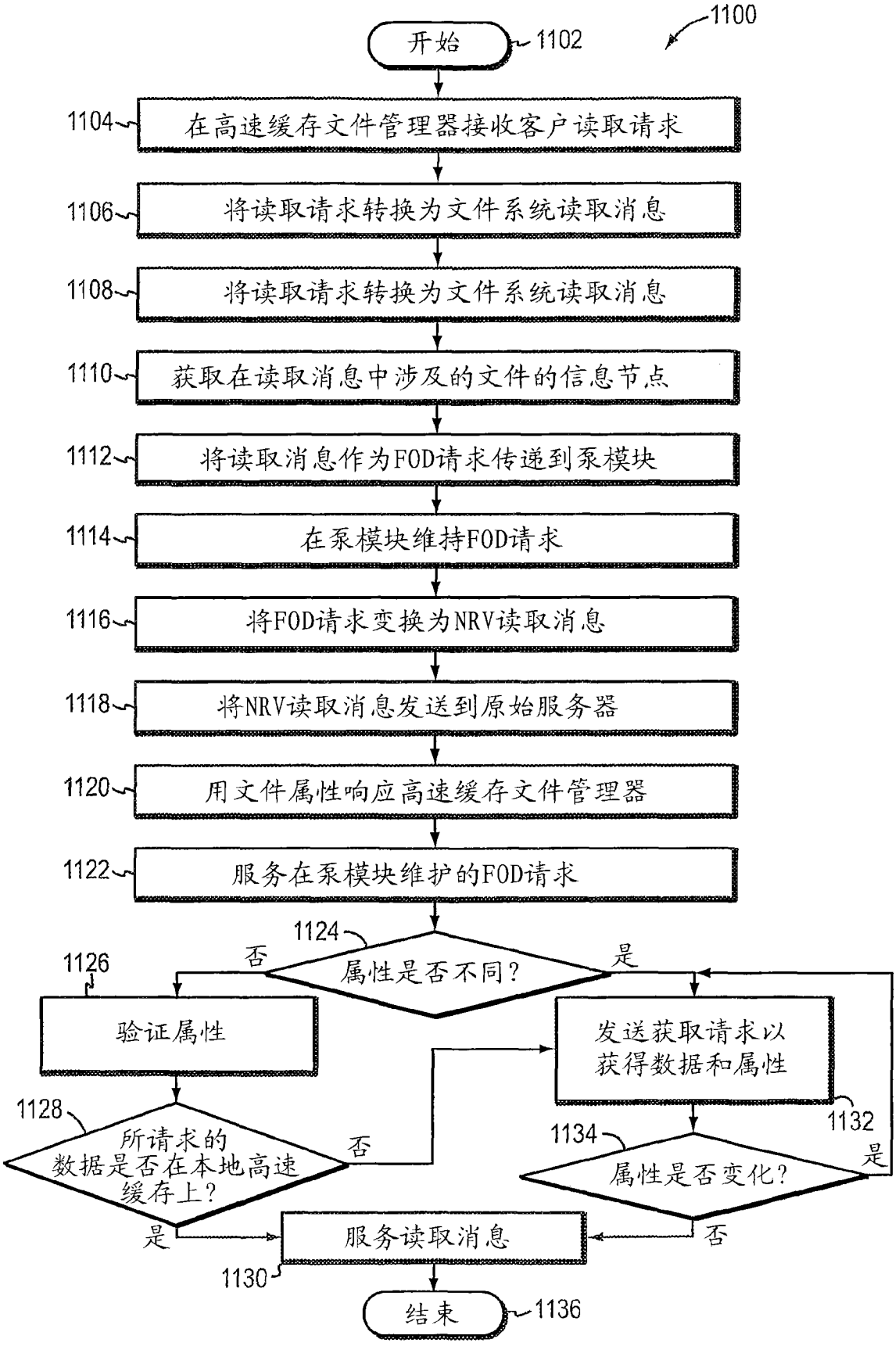

1100
开始
1102
1104
在高速缓存文件管理器接收客户读取请求
1106
将读取请求转换为文件系统读取消息
1108
将读取请求转换为文件系统读取消息
1110
获取在读取消息中涉及的文件的信息节点
1112
将读取消息作为FOD请求传递到泵模块
1114
在泵模块维持FOD请求
1116
将FOD请求变换为NRV读取消息
1118
将NRV读取消息发送到原始服务器
1120
用文件属性响应高速缓存文件管理器
1122
服务在泵模块维护的FOD请求
1124
属性是否不同?
否
是
1126
验证属性
发送获取请求以
获得数据和属性
1132
1128
所请求的
数据是否在本地高速
缓存上?
否
是
1134
属性是否变化?
是
否
服务读取消息
1130
结束
1136

图 11

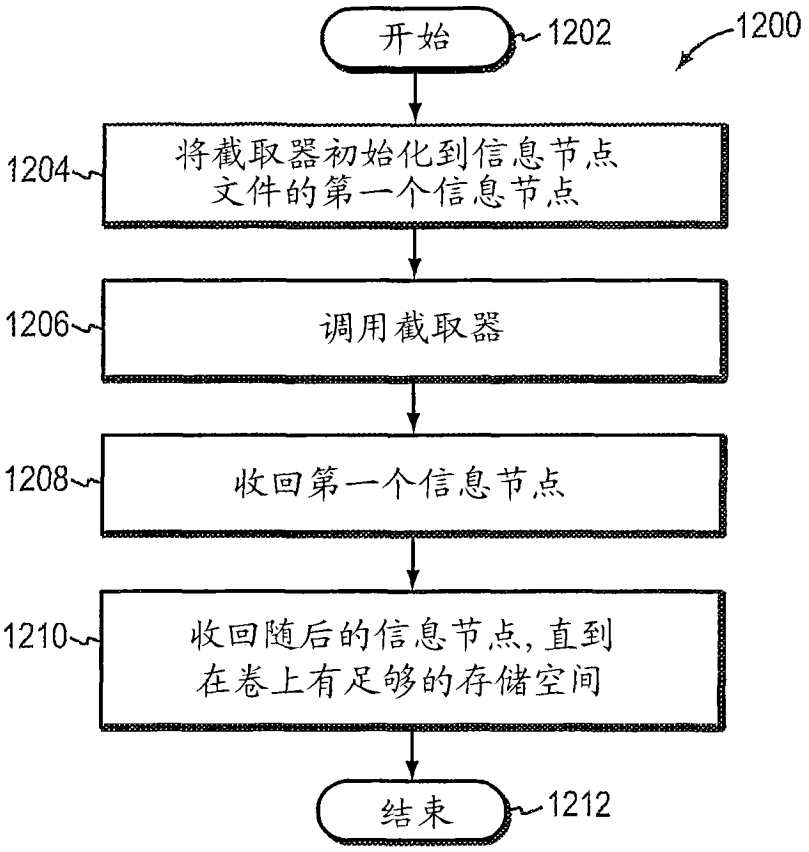

开始
1202
1200
1204
将截取器初始化到信息节点
文件的第一个信息节点
1206
调用截取器
1208
收回第一个信息节点
1210
收回随后的信息节点，直到
在卷上有足够的存储空间
结束
1212

图 12