(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 102971728 A
(43)申请公布日 2013.03.13

(21)申请号 201180021583.0
(22)申请日 2011.04.21
(30)优先权数据
12/770,577 2010.04.29 US
(85)PCT申请进入国家阶段日
2012.10.29
(86)PCT申请的申请数据
PCT/US2011/033478 2011.04.21
(87)PCT申请的公布数据
WO2011/139588 EN 2011.11.10
(71)申请人 赛门铁克公司
地址 美国加利福尼亚州
(72)发明人 S·S·曼莫汉 M·L·德希穆克
(74)专利代理机构 北京纪凯知识产权代理有限公司 11245
代理人 赵蓉民
(51)Int.Cl.
G06F 17/30(2006.01)
G06F 11/14(2006.01)

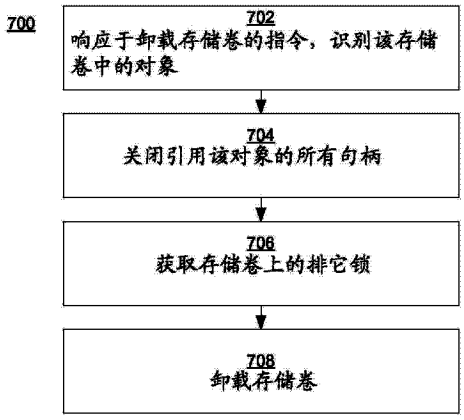
权利要求书 2 页 说明书 9 页 附图 7 页

(54)发明名称

卸载存储卷

(57)摘要

响应于卸载一个存储卷的指令，例如识别该存储卷中的一个对象并关闭引用该对象的一个句柄。一旦获取了该存储卷上的一个排它锁，就可以卸载该存储卷。然后可以重新安装该存储卷。



CN 102971728 A

1. 一种非瞬时性计算机可读存储媒质，该非瞬时性计算机可读存储媒质具有使一个计算机系统执行一种方法的计算机可读指令，该方法包括：

响应于卸载一个网络的第一节点中的一个存储卷的指令，识别所述存储卷中的一个对象；

关闭引用所述对象的一个句柄；

在所述关闭之后获取所述存储卷上的一个排它锁；并且

在获取所述排它锁之后卸载所述存储卷。

2. 如权利要求 1 所述的计算机可读存储媒质，其中所述方法进一步包括，在所述识别之前：

将缓存的数据从所述存储卷的一个文件缓冲器写入到所述存储卷；并且

尝试获取所述存储卷上的所述排它锁，其中如果所述尝试不成功，就执行所述识别和所述关闭。

3. 如权利要求 1 所述的计算机可读存储媒质，其中，所述关闭进一步包括：

访问每一个进程的一个内部句柄表，该进程正在访问所述存储卷，其中所述内部句柄表包括多个条目，该多个条目包括多个句柄；并且

识别引用所述存储卷的所述条目中的一个条目，其中所述条目对应于所述对象。

4. 如权利要求 3 所述的计算机可读存储媒质，其中所述识别一个条目包括：

识别所述条目中的对象的文件类型；

访问对象的所述文件类型的名称；并且

确定包含在所述名称中的文件路径是否对应于所述存储卷中的一个文件。

5. 如权利要求 4 所述的计算机可读存储媒质，其中所述方法进一步包括确定与所述文件相关联的访问权限。

6. 如权利要求 1 所述的计算机可读存储媒质，其中所述方法进一步包括在所述关闭之后和所述获取之前，将缓存数据从所述存储卷的一个文件缓冲器写入到所述存储卷。

7. 如权利要求 1 所述的计算机可读存储媒质，其中所述方法进一步包括，在所述卸载之后，重新安装所述存储卷。

8. 如权利要求 1 所述的计算机可读存储媒质，其中所述方法进一步包括在一个日志中指示所述句柄被关闭。

9. 一种包括计算机可读媒质的制品，该计算机可读媒质具有存储在其上的指令，如果由一个计算装置执行时，这些指令使所述计算装置执行多个操作，这些操作包括：

响应于第一次无法在一个网络的第一节点中获取一个存储卷上的一个排它锁的尝试，访问在所述计算装置上执行的每个进程的一个内部句柄表；

识别引用所述存储卷的所述内部句柄表中的一个条目；

关闭对应于所述条目的一个句柄；

在所述关闭操作之后，第二次尝试获取所述存储卷上的所述排它锁；并且

在获取所述排它锁之后卸载所述存储卷。

10. 如权利要求 9 所述的制品，其中所述操作进一步包括在所述第一次尝试之前将缓存数据从所述存储卷的一个文件缓冲器写入到所述存储卷。

11. 如权利要求 9 所述的制品，其中响应于卸载所述存储卷的一条指令来进行所述第

一次尝试。

12. 如权利要求 9 所述的制品，其中所述识别一个条目的操作包括：

识别所述内部句柄表中的对象的文件类型；

访问对象的所述文件类型的名称；以及

确定包含在所述名称中的文件路径是否对应于所述存储卷中的一个文件。

13. 如权利要求 12 所述的制品，其中所述操作进一步包括确定与所述文件相关联的访问权限。

14. 如权利要求 9 所述的制品，其中所述操作进一步包括在所述关闭操作之后和所述第二次尝试之前，将缓存数据从所述存储卷的一个文件缓冲器写入到所述存储卷。

15. 如权利要求 9 所述的制品，其中所述操作进一步包括在所述卸载之后重新安装所述存储卷。

16. 一种计算机系统，包括：

一个处理器；以及

存储器，该存储器连接至所述处理器并具有存储在其上的指令，如果由一个计算机系统执行时，这些指令使所述计算机系统执行一种，该方法包括：

将一个存储卷的一个文件缓冲器中的数据写入到所述存储卷；

在获取所述存储卷上的一个排它锁之前：

确定一个进程正在访问所述存储卷；

访问所述进程的一个内部句柄表；

识别引用所述存储卷的所述内部存储表中的一个条目；

关闭对应于所述条目的一个句柄；

在所述关闭之后，获取所述排它锁；并且

在获取所述排它锁之后卸载所述存储卷。

17. 如权利要求 16 所述的计算机系统，其中所述识别一个条目包括：

识别所述内部句柄表中的对象的文件类型；

访问对象的所述文件类型的名称；并且

确定包含在所述名称中的文件路径是否对应于所述存储卷中的一个文件。

18. 如权利要求 17 所述的计算机系统，其中所述方法进一步包括确定与所述文件相关联的访问权限。

19. 如权利要求 16 所述的计算机系统，其中所述方法进一步包括在所述关闭之后和获取所述排它锁之前将所述文件缓冲器中的数据写入到所述存储卷。

20. 如权利要求 17 所述的计算机系统，其中所述方法进一步包括在所述卸载之前重新安装所述存储卷。

卸载存储卷

背景技术

[0001] 在集群网络(例如,服务器集群)中,多个实体(例如计算机系统以及在那些系统上运行的应用程序)都可以访问相同的存储卷。那些应用程序中的一些(本文称为主应用程序)使用率较高。其他的应用程序本文称为辅应用程序。

[0002] 出于各种原因中的任意一种,主应用程序可能变得不可利用。例如,服务器(应用程序在其上执行)可能变得不可利用,在这种情况中应用程序也是不可用的,直到恢复服务器的服务。为了解决这种情况,尽可能快地在不同的服务器上重新启动应用程序,这个过程称为故障转移。

[0003] 为了从一个服务器向另一个服务器转移主应用程序的执行,要卸载该应用程序所使用的存储卷(本质上,是使存储卷脱机),然后重新安装(本质上,使它联机并再次用于该应用程序,当前就在第二个服务器上执行)。为了卸载存储卷,应当获取存储卷上的排它锁。然而,如果辅应用程序中的一个或多个继续访问存储卷,就不可能获取排它锁。

[0004] 按照惯例,在这种情况下强制性地卸载共享的存储器。这通常会在文件系统中产生非一致性数据。因此,管理员不得不手动地或自动地运行一种程序(例如“chkdsk”)以识别并修复错误。如果 chkdsk 不能够修复存储卷,则存储卷就无法重新安装,这会增加停机时间并因此降低了主应用程序的可用性。

[0005] 在一些群集网络实施方式中,集群软件可用于控制主应用程序的启动和停止,这样主应用程序不会影响获取排它锁的尝试。然而,辅应用程序不受集群软件的控制,因此在存储卷上可能具有打开的文件。在这些情况下,辅应用程序可能无法从存储卷卸载或者强制性地进行卸载,这可能导致写入错误,而写入错误进而可能导致文件系统不一致。

发明内容

[0006] 在一个实施方案中,响应于卸载一个存储卷的一条指令,识别该存储卷中的一个对象并关闭引用该目标的一个句柄。一旦获取该存储卷上的一个排它锁,就卸载该存储卷。

[0007] 在另一个实施方案中,响应于第一次无法在一个存储卷上获取一个排它锁的尝试,访问一个进程的一个内部句柄表。识别引用该存储卷的内部句柄表中的一个条目,并且关闭对应于该条目的一个句柄。然后,又一次尝试获取该存储卷上的排它锁。如果成功,则卸载该存储卷。

[0008] 更具体地,在一个实施方案中,对卷文件缓冲器进行刷新,这样所有的缓存数据被写入到存储卷。然后,尝试获取存储卷上的排它锁。如果尝试获取排它锁失败,这表示存在仍然正访问存储卷的至少一个应用程序 / 进程,在这种情况中执行以下的操作。访问系统中每个进程的内部句柄表,以便识别该进程所打开的对象。对于每个进程的内部句柄表中的每个条目,识别该条目所表示的对象的类型。具体而言,识别被标识为“文件”类型的条目。对于对象的每个文件类型,确定该对象的名称(例如,该打开的文件句柄的文件路径)。在一个实施方案中,还确定授权的访问权限(例如,读模式、写模式、或读 / 写模式)如果对象的名称对应于即将卸载的存储卷上的一个文件,则代表应用程序 / 进程来关闭该打开的句

柄。可以为可能的未来参考来记录以上所提的校正措施。接着，再次刷新卷文件缓冲器，并且再次尝试获取存储卷上的排它锁。刚刚描述的操作可以重复，直到获取到了排它锁。一旦获取了存储卷的排它锁，就卸载该卷。然后，重新安装该卷。

[0009] 因此，根据本披露的实施方案，在卸载一个共享卷之前停止主应用程序，这样可以在一个节点 / 系统上停止一个应用程序并在另一个节点 / 系统上重新启动它，而不会损害文件系统。在卸载该存储卷之前并且在停止主应用程序之后，如果存在访问该卷的任意辅应用程序 / 进程，则关闭它们的活动句柄并刷新该卷文件缓冲器，这样在故障转移之后该文件系统仍保持一致。根据本披露的实施方案可用于物理的和虚拟的环境。除了故障转移外，根据本披露的实施方案可以用于灾难恢复。

[0010] 本领域的普通技术人员在阅读不同附图中示出的实施方案的详细描述之后将会认识到本披露的不同实施方案的这些和其他目标以及优点。

附图说明

[0011] 结合在本说明书中并构成了它的一部分并且其中相似的数字描绘相似的元件的附图展示了本披露的实施方案并且与说明书共同用于解释本披露的原理。

[0012] 图 1 的方框图描绘了一种群集网络的实例的元件，根据本发明的实施方案可以在该群集网络上实施。

[0013] 图 2 的方框图描绘了一种计算机系统的实例，根据本发明的实施方案可以在计算机系统上实施。

[0014] 图 3A、3B 以及 3C 的方框图描绘了根据本发明的一种实施方案的存储卷的卸载 / 安装。

[0015] 图 4 的方框图描绘了根据本发明的一种实施方案的处理过程以及相关的句柄表。

[0016] 图 5 是根据本发明的一种实施方案的句柄表的实例。

[0017] 图 6 是根据本发明的一种实施方案的卸载和安装存储卷的计算机实施过程的流程图。

[0018] 图 7 是根据本发明的一种实施方案的卸载存储卷的计算机实施过程的流程图。

[0019] 图 8 是根据本发明的另一种实施方案的卸载和安装存储卷的计算机实施过程的流程图。

具体实施方式

[0020] 现在详细参考本披露的不同实施方案，在附图中示出了这些实施方案的实例。当结合这些实施方案进行描述时，应当理解的是它们无意将本披露限制于这些实施方案。相反，本披露意在涵盖多种替代形式、修改形式以及等同形式，它们包括在所附权利要求书定义的本披露的精神和范围内。而且，在本披露的以下详细描述中，给出了很多具体细节以提供对本发明的透彻理解。然而，应当理解的是在不具有这些特定细节的情况下可以实施本披露。在其他实例中，并未详细描述众所周知的方法、过程、组件以及电路，以免不必要地混淆本发明的多个方面。

[0021] 下文的详细说明的一些部分描述在计算机存储器中数据位上的操作的流程、逻辑块、处理、及其他符号图示。这些说明和图示是数据处理领域中那些熟练技术人员向该领域

的其他熟练技术人员更有效地传达他们工作的实质所使用的手段。在本申请中，程序、逻辑框、过程等被构想为产生所需结果的步骤或指令的一个前后一致序列。这些步骤是利用对物理量进行物理操作的那些步骤。通常，尽管不是必要的，这些量采用能够在计算机系统中存储、转移、组合、比较、及以其他方式处理的电或磁信号的形式。主要出于常用的原因，已经证明参考如事务、位、值、元素、符号、字符、样本、像素等等的这些信号有时是方便的。

[0022] 然而应记住，所有这些或类似术语与合适的物理量相关联并且仅是应用到这些量的方便的标签。除非在其他方面另有说明(如从以下讨论中显而易见的)，应当认识到贯穿本披露，利用术语例如“识别”、“关闭”、“获取”、“卸载”、“安装(重新安装)”、“命名(重新命名)”、“写入”、“尝试”、“访问”、“确定”、“记录”、“致使”、“指示”等进行的讨论是指计算机系统或类似电子计算装置或处理器(例如，图 2 的系统 210)的动作或进程(例如，分别是图 6、7 和 8 的流程图 600、700 和 800)。该计算机系统或类似的电子计算装置操纵和转换表示为在计算机系统存储器、寄存器或其他这样的信息存储、传递或显示装置中的物理(电子)量的数据。

[0023] 本文描述的实施方案可以在计算机可执行指令的一般环境中进行讨论，这些指令驻留于由一个或多个计算机或其他装置执行的某种形式的计算机可读存储媒质(例如程序模块)上。通过示例，并且不限制，计算机可读存储媒质可以包括计算机存储媒质和通信媒质。总体上，程序模块包括执行具体任务或实施具体或抽象数据类型的历程、程序、目标、组件、数据结构等。程序模块的功能可以如在各种实施方案中所期望的一样组合或分布。

[0024] 计算机存储媒质包括在用于信息(例如计算机可读指令、数据结构、程序模块、或其他数据)存储的任何方法或技术中实施的易失性和非易失性、可移动和非可移动媒质。计算机存储媒质包括但不限于随机存取存储器(RAM)、只读存储器(ROM)、电可擦可编程 ROM (EEPROM)、闪存存储器、或其他的存储器技术、高密度磁盘 ROM (CD-ROM)、数字视频光盘(DVD)或其他的光存储器、磁带盒、磁带、磁盘存储器或其他磁存储装置、或可以用来存储所期望的信息并且可以对其进行访问以检索那些信息的任何其他媒质。

[0025] 通信媒质可以包括计算机可执行指令、数据结构以及程序模块，并且包括任意信息传送媒质。通过示例而不是不限制，通信媒质可以包括有线媒质(例如有线网络或直接有线连接)以及无线媒质(例如声学、射频(RF)、红外线及其他无线媒质)。上述任何内容的组合还可以包括在计算机可读存储媒质的范畴中。

[0026] 总体而言，根据本披露的实施方案提供了多种方法和系统以便平和地卸载和安装共享的存储卷，这样可以在不损害文件系统的情况下将应用程序的执行从一个系统或节点转移到另一个系统。在卸载该卷之前，如果存在访问该卷的任何应用程序，则关闭它们的活动句柄并刷新卷文件缓冲器以便文件系统在故障转移之后仍保持一致。

[0027] 图 1 的方框图描绘了群集网络 100 的实例的元件，在该网络上可以实施根据本发明的实施方案。在图 1 的实例中，该网络包括 N 个(例如 32 个)节点或系统(例如计算机系统或服务器)，每个节点或系统连接至共享储存器 130。网络 100 可以称为高可用性集群(HAC)，通过提供将那些服务从一个节点向另一个转移(例如，故障转移)以响应于计划事件(例如，用于维护、升级、打补丁)或意外事件(例如，节点意外地变得无效)的服务能力，该集群提高了服务(例如应用程序)的可用性。那些服务可以包括例如数据库、文件共享以及电子商务。

[0028] 为了获取高可用性，集群软件可以用于监测应用程序和节点 / 系统的状态，并自动地将应用程序的执行从一个系统向另一系统转移以响应计划或意外事件。在图 1 的实例中，集群软件包括运行在 N 个系统的每一个上的代理 112。代理 112 可以监测每个系统上运行的应用程序并可以触发与启动、停止和转移应用程序（由集群软件来监测和控制）的执行相关联的操作。

[0029] 并不是所有的在系统 1、……、N 上执行的应用程序可以由集群软件进行监测和 / 或控制。为了便于讨论，由集群软件进行监测和控制的应用程序在本文称为主应用程序，而其他应用程序称为辅应用程序。例如，主应用程序可以是高可用性应用程序（需要或希望高度可用性的应用程序）。

[0030] 在图 1 的实例中，共享存储器 130 是数据存储系统，该系统可以包括一个或多个物理存储装置，例如一个或多个物理磁盘、LUN（小型计算机系统接口（SCSI）逻辑单元）或用于存储数据的其他类型的硬件。存储卷 132 代表一个或多个数据卷（块），在卷管理器（未示出）的控制下，这些数据卷（块）可以位于一个单一的物理磁盘上或可以分布在多个物理磁盘（称为虚拟磁盘）上。存储卷 132 可以包括卷文件缓冲器 131，在数据写入到存储器 132 之前该缓冲器刷新数据。存储卷 132 可以由系统 1 上执行的主应用程序 110 和辅应用程序 111 进行访问。

[0031] 图 2 描绘了适于执行本披露的计算机系统 210 的方框图。在以下讨论中，描绘了不同的而且很多的组件和元件。这些组件的各种组合形式和子集可用于实现结合图 1 所提及的装置。例如，系统 1、……、N 每一个可以是全功能计算机系统（如果不是计算机系统 210 的全部特征的话，该系统也采用了很多），或它们可以仅使用支持功能（由那些装置提供）所需的那些特征的子集。例如，服务器可以不需要键盘或显示器，并且可以执行相对稀疏的支持数据存储和数据访问的功能以及这类功能的管理的操作系统。

[0032] 在图 2 的实例中，计算机系统 210 包括使计算机系统的主要子系统相互连接的总线 212。这些子系统包括中央处理器 214；系统存储器 217；输入 / 输出控制器 218；外部音频装置，例如经过音频输出接口 222 的扬声器系统 220；外部装置，例如经过显示适配器 226 的显示屏 224；串行接口 228 和 230；键盘 232（与键盘控制器 233 相连接）；存储接口 234；可操作用于接收软盘 238 的软盘驱动器 237；可操作用于连接光纤通道网络 290 的主机总线适配器（HBA）接口卡 235A；可操作用于连接至 SCSI 总线 239（SCSI 的替代形式包括集成开发环境（IDE）和串行高级技术附件（SATA））的 HBA 接口卡 235B；以及可操作用于接收光盘 242 的光盘驱动器 240。还包括鼠标 246（或其他点击装置，它通过串行端口 228 连接至总线 212）；调制解调器 247（通过串行端口 230 连接至总线 212）；以及网络接口 248（直接连接至总线 212）。调制解调器 247、网络接口 248 或一些其他方法可用于提供图 1 的网络 100 的连通性。

[0033] 图 2 的总线 212 允许中央处理器 214 和系统存储器 217 之间的数据通信，系统存储器可以包括如前文提及的 ROM 或闪存以及 RAM（未示出）。RAM 通常是加载操作系统和应用程序的主存储器。除其他代码之外，ROM 或闪存可以包含基本输入输出系统（BIOS），该系统控制基本的硬件操作，例如与外围组件的交互。

[0034] 驻留在计算机系统 210 中的应用程序通常存储在一种计算机可读存储媒质上并且可以通过该计算机可读媒质来访问这些应用程序，例如硬盘驱动器（如固定盘 244）、光盘

驱动器(如光盘驱动器 240)、软盘单元 237、或者其他存储媒质。当通过网络调制解调器 247 或接口 248 进行访问时,应用程序可以是根据应用程序和数据通信技术进行调制的电信号的形式。

[0035] 继续参见图 2,如同计算机系统 210 的其他存储器接口一样,存储器接口 234 可以连接至标准计算机可读存储媒质以进行信息的存储和 / 或检索,例如固定盘驱动器 244。固定盘驱动器 244 可以是计算机系统 210 的一部分或它可以是分离的并通过其他接口系统来访问。调制解调器 247 可以通过一个电话链路提供到远程服务器上的直接连接,或者通过互联网服务提供商(ISP)提供到互联网的直接连接。网络接口 248 可以通过一个直接网络链路提供到一个远程服务器的直接连接,或者通过一个 POP (存在点)提供到互联网的直接连接。网络接口 248 可以使用无线技术提供此类连接,包括数字蜂窝电话连接、蜂窝数字包数据(CDPD)连接、数字卫星数据连接或类似的连接。

[0036] 许多其他装置或子系统(未在图 2 中示出)能够以类似的方式(例如,文档扫描仪、数码照相机等)进行连接。相反,实施本披露并不要求图 2 中示出的所有装置都存在。这些装置和子系统能够以不同于图 2 中示出的方式互相连接。

[0037] 如图 2 中所示的计算机系统的操作在本领域中是公知的并且未在本申请中进行详细讨论。实施本披露的代码可以存储在计算机可读存储媒质上,例如系统存储器 217、固定盘 244、光盘 242 或软盘 238 中的一个或多个。计算机系统 210 上提供的操作系统可以是 MS-DOS®、MS-WINDOWS®、OS/2®、UNIX®、Linux®或另一种已知的操作系统。

[0038] 另外,就本文描述的信号而言,本领域熟练技术人员将认识到,信号可以从第一模块直接传递到第二模块,或可以在模块之间修改(例如,放大、衰减、延迟、锁存、缓冲、反转、滤波或以其他方式进行修改)信号。尽管以上所述实施方案的信号的特征为从一个模块向下一个模块传输,但只要信号的信息和 / 或功能方面在模块之间进行传输,本披露的其他实施方案就可以包括修改的信号以替代这种直接传输的信号。在某种程度上,在第二模块上输入的信号可以概念化为来源于第一信号的第二信号,而该第一信号输出自第一模块,这是因为所涉及的电路的物理限制(例如,不可避免地存在一些衰减和延迟),因此,如本文所用,源自第一信号的第二信号包括第一信号或第一信号的任意修改形式,是由于电路限制还是由于经过其他电路元件并不会改变第一信号的信息和 / 或最终功能方面。

[0039] 如以上所提,根据本披露的实施方案提供多种方法和系统,以便平和地卸载和安装共享的存储卷,因此应用程序可以在一个系统上停止并在另一个系统上安装,并不会损害文件系统。在以下讨论指代一个单一应用程序的情况下,这种讨论可以方便地扩展至多个应用程序。

[0040] 首先参见图 3A,主应用程序 310 和辅应用程序 311 在系统 1 上执行。在图 3A 的实例中,应用程序 310 和 311 可以将数据从存储卷 132 (例如,卷名称“F:\”)读取数据和 / 或从其写入。特别重要的是将数据写入存储卷 132 的那些应用程序,但本披露不限于那种类型的应用程序。注意,当共享存储卷 132 安装在系统 1 上时应用程序 330 和 331 不能访问它,一旦共享存储卷 132 从系统 1 卸载并安装在系统 N 上时,这些应用程序就可以访问它。

[0041] 在图 3B 的实例中,决定停止在系统 1 上执行主应用程序 310 并开始在系统 2 上执行冗余的主应用程序 310。换言之,决定将主应用程序 310 从系统 1 向系统 2 进行故障转

移。如上所提及，这种决定可以是计划的或意外的运行中断的结果。为了对主应用程序 310 进行故障转移，关闭该程序以及使用存储卷 132（由主应用程序 310 使用的相同存储卷）的其他应用程序，因此随后可以卸载存储卷 132。

[0042] 主应用程序 310 可以通过集群软件（例如，图 1 的代理 112）来关闭。然而，如前文所提及，辅应用程序 311 不受集群软件的控制。为了关闭辅应用程序 311，在一个实施方案中，使用以下结合图 6 描述的方法。

[0043] 继续参见图 3B，在关闭访问存储卷 132 的主和辅应用程序之后，可以卸载存储卷 132。一旦执行了那些操作，主应用程序 310 可以从系统 1 向系统 2 进行故障转移。

[0044] 现在参见图 3C，在重新安装存储卷 132 之后，主应用程序 310 可以故障转移至系统 2。更具体地，在图 3C 的实例中，存储卷 132 在系统 2 上作为“F:\”被重新安装，并且在系统 2 上启动主应用程序。因此，存储卷 132 可以由运行在系统 2 上的任何其他应用程序通过主应用程序 310（它当前正在系统 2 上执行）进行访问。

[0045] 通常来说，运行在系统 1、2、……、N 上的应用程序的每一个与一个或多个进程相关联。现在参见图 4 的实例，进程 1 与辅应用程序 311 相关联，进程 2 与一个不同的应用程序相关联，而进程 M 与又一个应用程序相关联。当本讨论指代一个单一进程时，这种讨论可以容易地扩展到多个进程。

[0046] 内部句柄表与每个进程相关联。一般来说，句柄表是与一个具体进程相关联的操作系统专用数据结构并且识别（列出）该进程打开的对象（例如，执行对象）。通常，每个系统 / 节点有一个句柄表。在图 4 的实例中，句柄表 1 与进程 1 相关联，进程 1 与在系统 1 上执行的辅应用程序 311（图 3A）相关联。类似地，句柄表 2 与进程 2 相关联，进程 2 与群集网络 100（图 1）中的多个系统之一（或者是系统 1 或者是另一个系统）上执行的不同应用程序相关联，以此类推。

[0047] 在一个实施方案中，每个句柄表包含很多条目，并且每一个条目识别一个打开的句柄。例如，句柄表 1 包括条目（打开的句柄）1、2、……、K。句柄表中用于一个进程的每个句柄对应于与该进程相关联的一个对象（例如，执行对象）。因此，句柄表 1 包括与进程 1 相关联的对象列表。

[0048] 如图 4 中所示，用于一个对象的条目包括但不限于例如与该对象相关联的唯一名称、对象类型、以及与该对象相关联的权限。从以下讨论中将会看到，类型为“文件”的对象是尤其令人感兴趣的。

[0049] 图 5 提供了根据本发明的一种实施方案的信息类型的实例，该信息可以包括在句柄表 1 中。句柄表 1 包括对象的完整名称、对象类型以及相关权限。对于对象的文件类型，完整的名称包括完整的文件路径，文件路径包括文件所驻留的存储卷的标识。在图 5 中，未示出完整的名称 / 文件路径；只示出了安装的卷的名称。例如，句柄 1 与映射至“F:\”（图 1 的共享存储器 130 上的存储卷 132）的对象的文件类型相关联，而句柄 4 与映射至不同存储卷“C:\”（图中未示出）的对象的文件类型相关联对于对象的文件类型，权限类型包括只读、只写以及读 / 写（读和写）。

[0050] 图 6 是根据本发明的一种实施方案的用于卸载和重新安装存储卷的计算机实施处理过程的流程图 600。图 7 是根据本发明的一种实施方案的用于卸载存储卷的计算机实施处理过程的流程图 700。图 8 是根据本发明的另一种实施方案的用于卸载存储卷的计算

机实施处理过程的流程图 700。流程图 600、700 和 800 可以实施为计算机可执行指令，这些指令驻留在某种形式的计算机可读存储媒质（例如，位于图 2 的系统 210 中）上。

[0051] 在图 6 的模块 602，在一个实施方案中，刷新卷文件缓冲器 131（图 1），这样所有的缓存数据（例如，新技术文件系统（NTFS）数据）被写入到图 1 的存储卷 132（例如，名为“F:\”的存储卷）。这种功能可以响应于操作系统提供的应用程序接口（API）所发出的命令或指令来执行。

[0052] 在模块 604，在一个实施方案中，尝试获取存储卷 132（F:\）上的排它锁。这种功能可以通过操作系统提供的另一个 API 来执行。

[0053] 在模块 606，如果可以获取排它锁，流程图进行到模块 618。然而，如果尝试获取排它锁失败，这表示至少有一个应用程序 / 进程仍在访问存储卷 132(F:\)，在这种情况下，流程图 600 进行到模块 608。模块 604 中所用的 API 可以发出一条命令或指令，以指示它不能够获取排它锁。

[0054] 在模块 608，访问存储卷 132 所在的系统上的每个进程的内部句柄表（图 4 和 5）以便识别各自进程所打开的对象（例如，文件）。也就是说，访问系统 1 上的一个进程所对应的每个句柄表（图 4）以确定进程中的任意一个是否正在访问存储卷 132（F:\）。

[0055] 在模块 610，对于系统上每个进程的内部句柄表中的每个条目（打开的句柄），对条目所代表的对象的类型进行识别。具体而言，对识别为“文件”类型的条目（句柄）进行标识（见图 5）。在一个实施方案中，只针对类型为文件的对象搜索句柄表。对象的文件类型是尤其令人感兴趣的，因为那些对象是访问（读取和 / 或写入）存储卷 132（F:\）的对象。

[0056] 在模块 612，对于模块 610 中所识别对象的每个文件类型，确定对象（例如，打开的文件句柄的文件路径）的名称（见图 5）。如果对象的文件类型具有映射至存储卷 132（F:\）的名称（文件路径），则关闭该句柄。在一个实施方案中，还确定授权的访问权限（例如，读模式、写模式、或读 / 写模式）。

[0057] 在模块 614，如果对象的名称对应于即将卸载的存储卷 132 上的一个文件，则代表应用程序 / 进程来关闭该对象的打开句柄。如果除了即将卸载的存储卷外，进程正在访问其他存储卷上的文件，则只关闭被卸载的存储卷（存储卷 132）上的那些对象的句柄，从而使其他存储卷上的其他文件的句柄打开。例如，参见图 5，句柄 1 和 3 被关闭，因为它们引用了存储卷 132（F:\），但没有必要关闭句柄 4（即使它与对象的文件类型相关联），因为它指向不同的存储卷（C:\）。操作系统提供的 API 可用于关闭适合的已打开句柄。

[0058] 出于某种原因，如果不能关闭一个句柄，则终止进程本身。如果终止了一个进程，则可以在卸载存储卷 132 之后重新启动它。

[0059] 在图 6 的模块 616，可以存录（记录）以上提及的相关动作以用于可能的未来参考。

[0060] 接着，流程图 600 返回到模块 602。换言之，再次刷新文件缓冲器 131，这样在执行刚刚描述的操作时积累的任意 NTFS 数据被写入到存储卷 132。在模块 604，又一次尝试获取存储卷 132 上的排它锁。如果不成功（例如，因为多个进程之一打开了一个新句柄），执行刚刚描述的操作，直到获取了排它锁。

[0061] 在模块 618，一旦为存储卷 132 获得了排它锁，就可以卸载该卷。

[0062] 在模块 620，安装（重新安装）存储卷 132，然后提供给故障转移系统 / 节点（例如，图 3C 的系统 2）。

[0063] 根据本发明的一个实施方案，存在三种可能的操作模式。在模式 1，如果没有获取存储卷 132 上的排它锁，那么集群软件中止使该卷脱机的尝试。然后模式 1 依赖于人工干预和校正步骤。在模式 2，集群软件以图 6 描述的方式平和地卸载存储卷 132。在模式 3，如果没有获取存储卷 132 上的排它锁，则强制性卸载该存储卷。

[0064] 现在参见图 7，在模块 702，响应于卸载存储卷的一条指令，识别存储卷 132（图 1）中的一个对象。在这个操作之前，并且响应于该卸载指令，可以将缓存数据从文件缓冲器 131（图 1）写入到存储卷 132，并可以尝试获取该卷上的排它锁。

[0065] 在模块 704，关闭引用该对象的句柄。在一个实施方案中，通过识别句柄表中引用该存储卷的条目而首次访问每个进程（正在访问存储卷 132）的内部句柄表（图 4 和 5）来关闭一个或多个句柄。更具体地，在一个实施方案中，识别句柄表中的对象的文件类型，访问对象的文件类型名称，并确定包括在名称中的文件路径是否对应于存储卷 132 中的文件。在一个实施方案中，上传一条日志以记录该打开的句柄被关闭。

[0066] 在模块 706，获取存储卷 132 上的排它锁。在获取排它锁之前，可以将缓存数据从文件缓冲器 131 再次写入到存储卷 132。

[0067] 接着在模块 708，可以卸载存储卷 132。在对应用程序进行故障转移之前，再次将卷 132 安装在不同的系统 / 节点上。

[0068] 现在参见图 8，在模块 802，响应于第一次无法获取存储卷上的排它锁的尝试，访问每个进程的内部句柄表。在该第一次尝试之前，可能响应于一条卸载指令，可以将缓存数据从文件缓冲器 131（图 1）写到存储卷 132（图 1）。

[0069] 在模块 804，识别内部句柄表（图 4 和 5）中的并且引用存储卷 132 的一个条目。更具体地，在一个实施方案中，如前文所描述的，识别句柄表中的对象的文件类型，访问对象的文件类型名称，并确定包括在名称中的文件路径是否对应于存储卷 132 中的文件。

[0070] 在模块 806，对应于该句柄表中的条目的句柄被关闭。

[0071] 然后，在模块 808，第二次尝试获取存储卷 132 上的排它锁。在进行第二次尝试之前，将缓存数据从文件缓冲器 131 再次写入到存储卷 132。

[0072] 在模块 810，如果成功获取排它锁，则卸载存储卷 132 并在随后安装（在应用程序的故障转移之前）。

[0073] 综上所述，根据本发明的实施方案，在可以卸载共享的存储卷之前停止主应用程序，这样一个应用程序可以停止并在另一个节点 / 系统上重新启动，而不会损害文件系统。在卸载存储卷之前和停止主应用程序之后，如果存在访问该卷的任意辅应用程序 / 进程，则关闭它们的活动句柄并刷新卷文件缓冲器，这样在故障转移之后文件系统仍保持一致。相对于传统方法，存储卷可以更快地卸载，因为辅应用程序的句柄很容易识别和关闭。因而，减少了停机时间并提高了主（高可用性）应用程序的可用性。

[0074] 根据本披露的实施方案可用在物理的和虚拟的环境中。除了故障转移，根据本披露的实施方案可以用于灾难恢复。

[0075] 为了进行解释，已经参照具体实施方案对前述说明作出了描述。然而，这些示意性的说明并不意味着穷举或者将本发明限制在所披露的准确形式。鉴于以上教导，许多修改和变形都是可能的。为了最好地解释本发明的原理及其实际应用，选择并说明了这些实施方案，从而使得本领域的其他技术人员能够最好地利用本发明，以及针对预期的具体用途

而作了各种适当修改的不同实施方案。

[0076] 因此，描述了根据本发明的实施方案。尽管本披露已经在具体实施方案中进行了描述，但应当认识到的是，本发明不应当在这些实施方案的限制的情况下进行解释，而是根据以下权利要求进行解释。

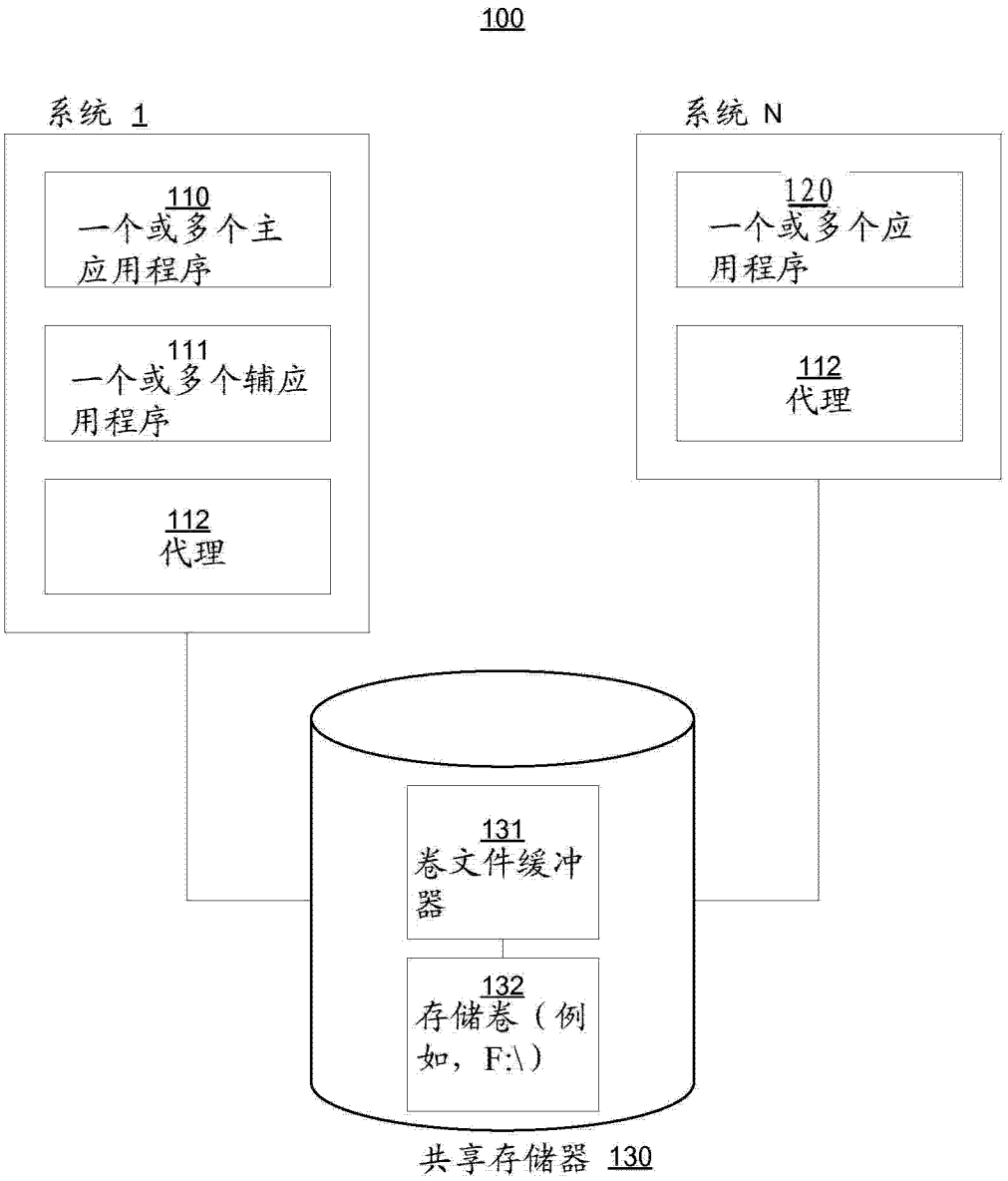

100
系统 1
110
一个或多个主应用程序
111
一个或多个辅应用程序
112
代理
系统 N
120
一个或多个应用程序
112
代理
131
卷文件缓冲器
132
存储卷（例如，F:\）
共享存储器 130

图 1

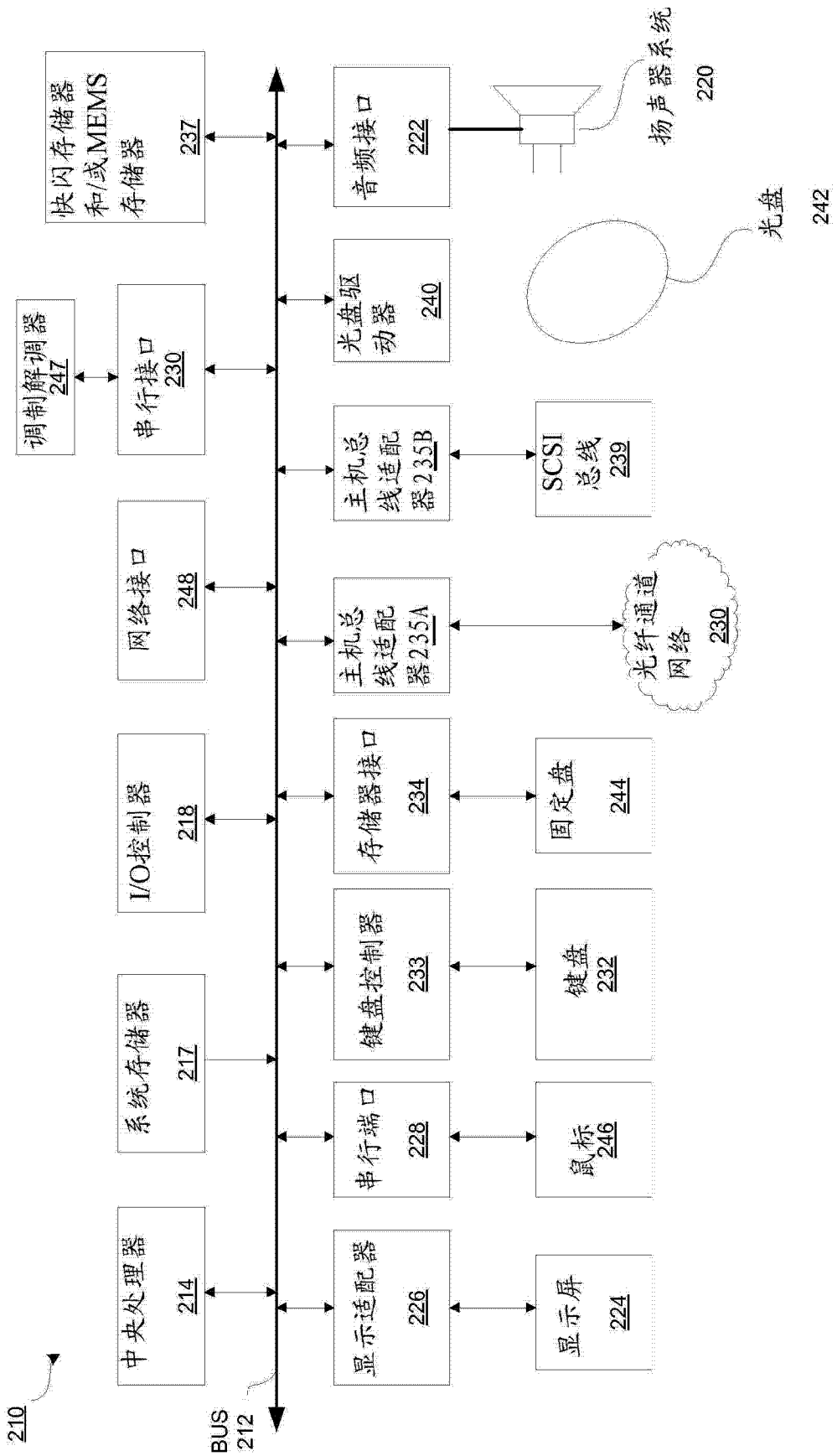

210
BUS 212
中央处理器 214
系统存储器 217
I/O控制器 218
网络接口 248
串行接口 230
调制解调器 247
显示适配器 226
串行端口 228
键盘控制器 233
存储器接口 234
主机总线适配器235A
主机总线适配器235B
光盘驱动器 240
音频接口 222
快闪存储器和/或MEMS存储器 237
显示屏 224
鼠标 246
键盘 232
固定盘 244
光纤通道网络 230
SCSI总线 239
光盘 242
扬声器系统 220

图 2

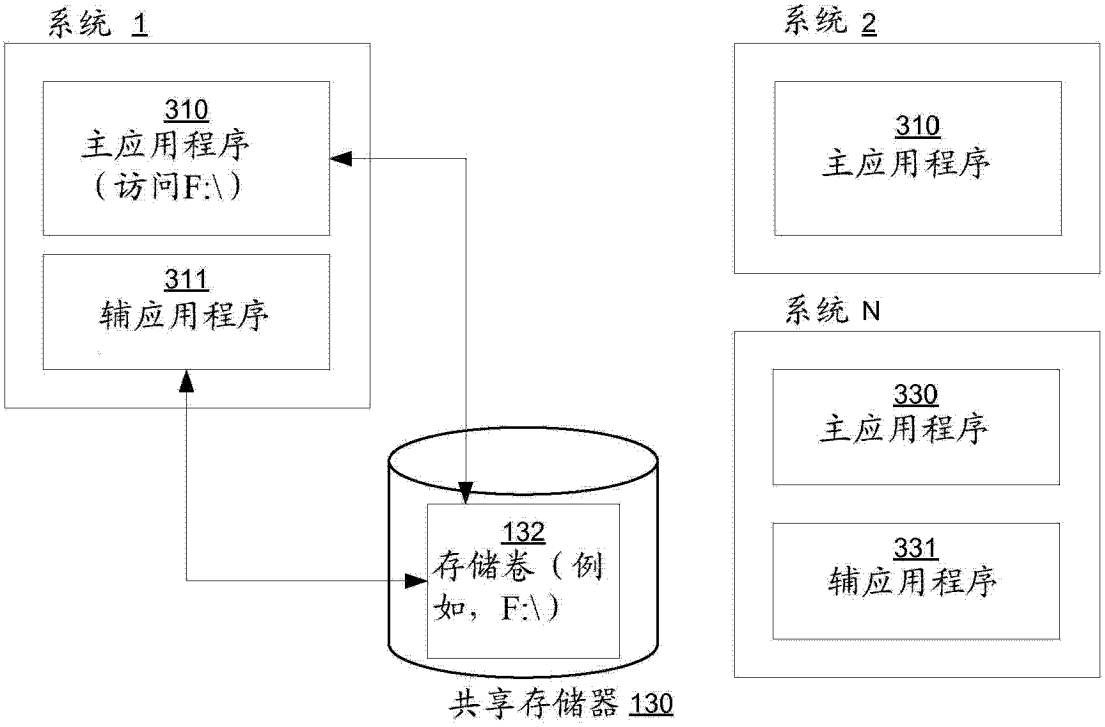

系统 1
310
主应用程序
（访问F:\）
311
辅应用程序
系统 2
310
主应用程序
系统 N
330
主应用程序
331
辅应用程序
132
存储卷（例如，F:\）
共享存储器 130

图 3A

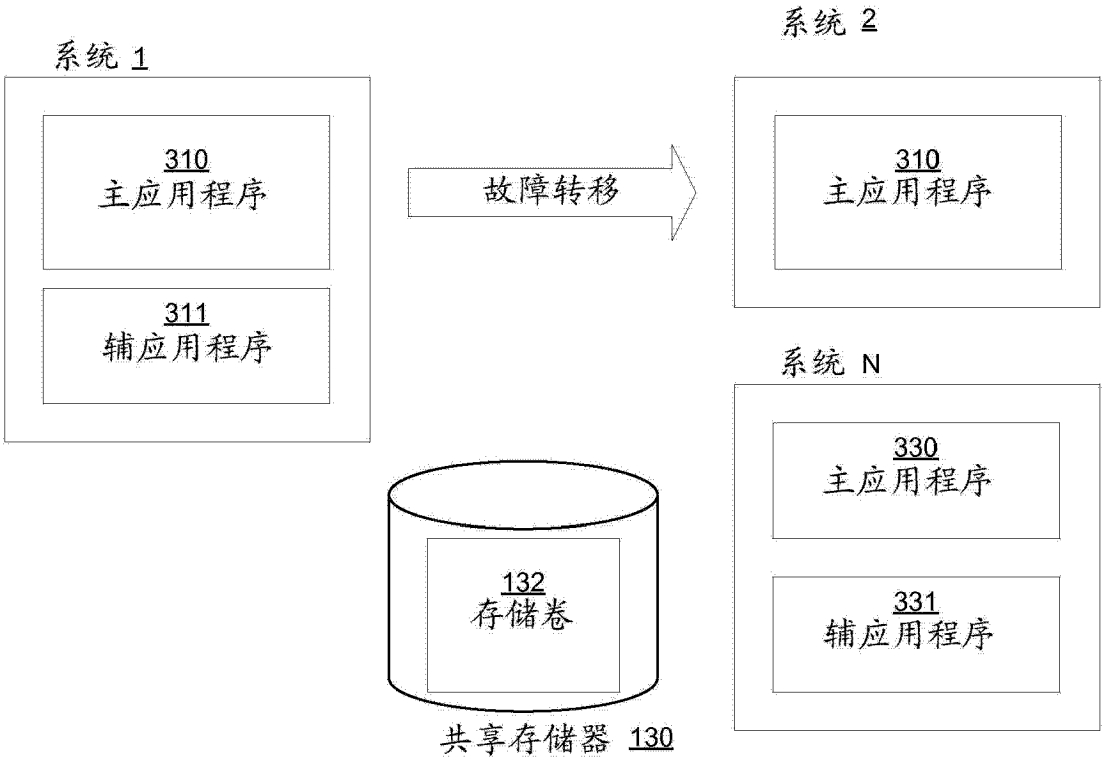

系统 1
310
主应用程序
311
辅应用程序
故障转移
系统 2
310
主应用程序
系统 N
330
主应用程序
331
辅应用程序
132
存储卷
共享存储器 130

图 3B

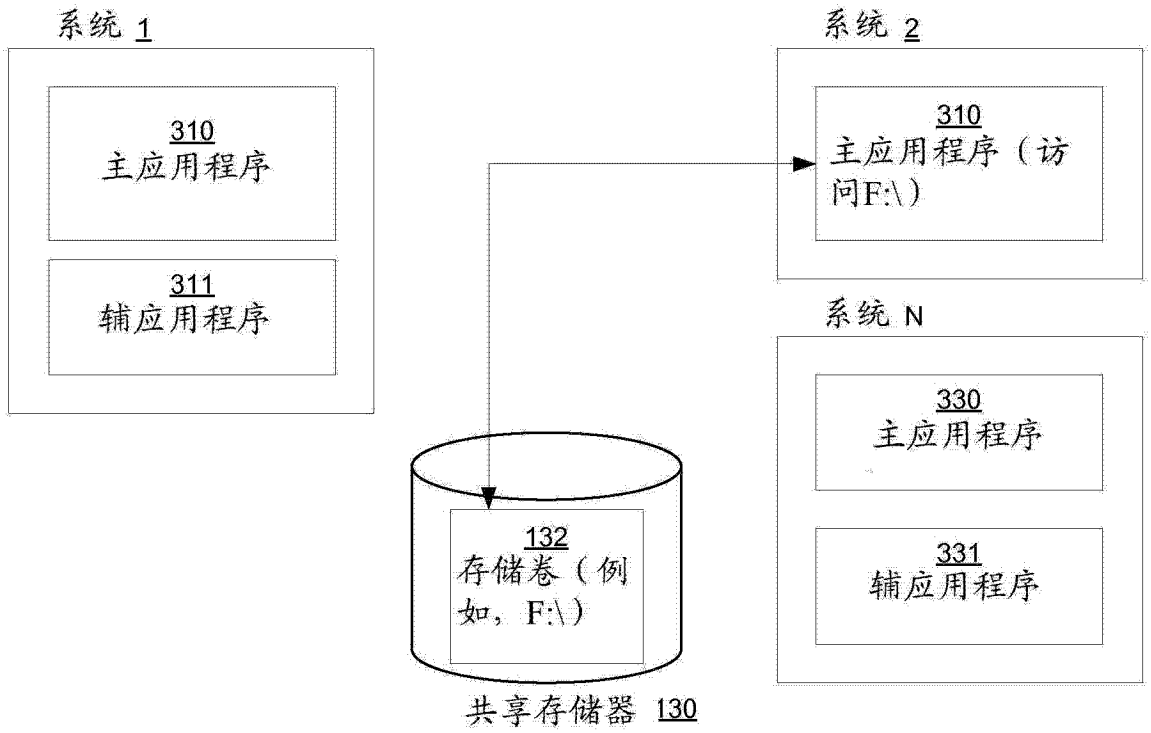

系统 1
310
主应用程序
311
辅应用程序
系统 2
310
主应用程序（访问F:\）
系统 N
330
主应用程序
331
辅应用程序
132
存储卷（例如，F:\）
共享存储器 130

图 3C

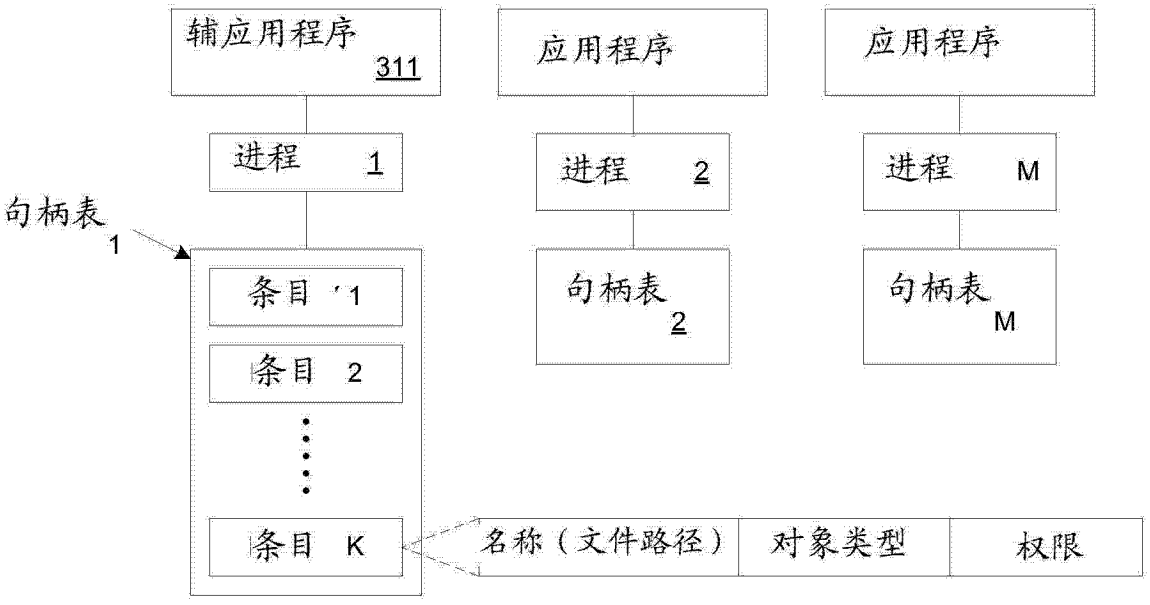

辅应用程序 311
应用程序
应用程序
进程 1
进程 2
进程 M
句柄表 1
条目 1
条目 2
条目 K
句柄表 2
句柄表 M
名称（文件路径）
对象类型
权限

图 4

句柄表

1

	名称（文件路径）	对象类型	权限
1	F:\	文件	读
2	HKLM/软件	不是文件	读/写
3	F:\	文件	写
4	C:\	文件	读/写

图 5

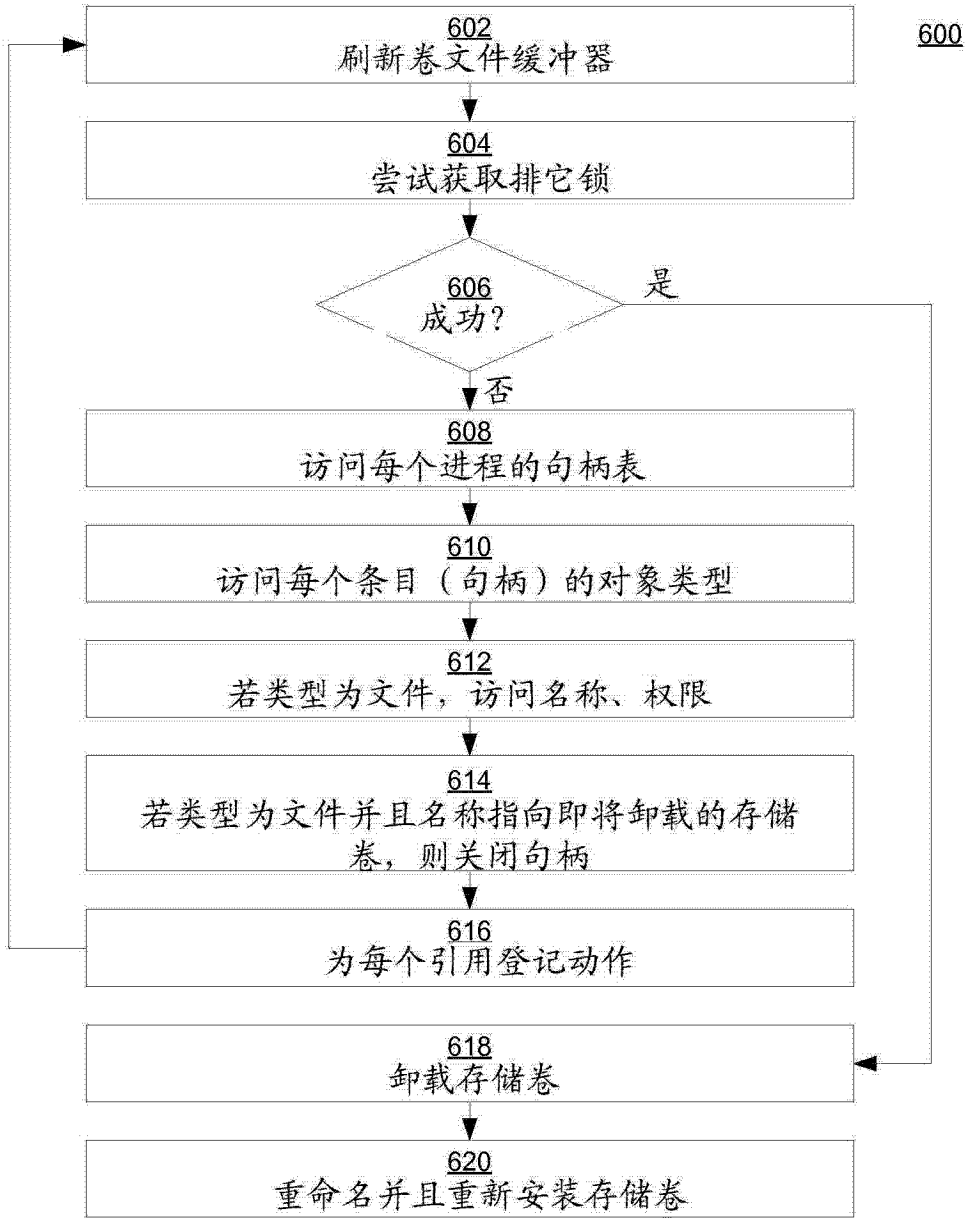

600
602
刷新卷文件缓冲器
604
尝试获取排它锁
606
成功?
是
否
608
访问每个进程的句柄表
610
访问每个条目（句柄）的对象类型
612
若类型为文件，访问名称、权限
614
若类型为文件并且名称指向即将卸载的存储卷，则关闭句柄
616
为每个引用登记动作
618
卸载存储卷
620
重命名并且重新安装存储卷

图 6

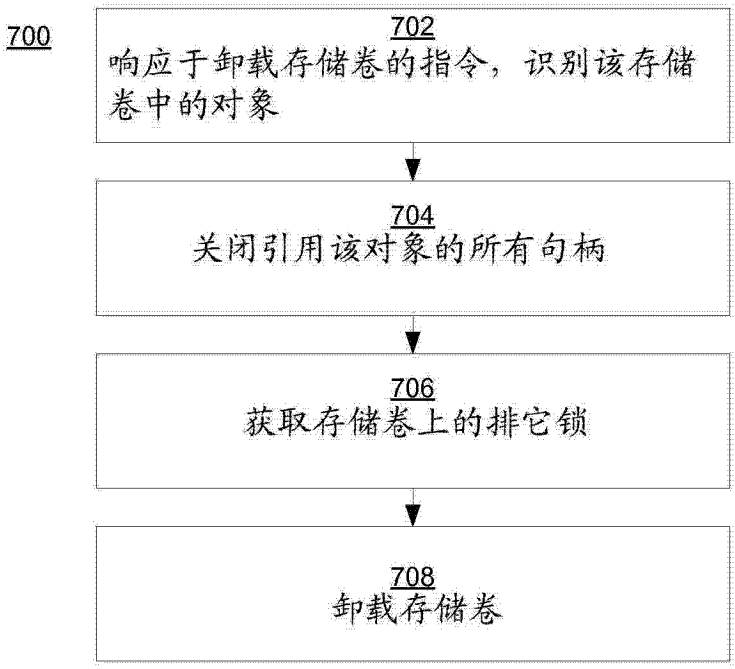

700
702
响应于卸载存储卷的指令，识别该存储卷中的对象
704
关闭引用该对象的所有句柄
706
获取存储卷上的排它锁
708
卸载存储卷

图 7

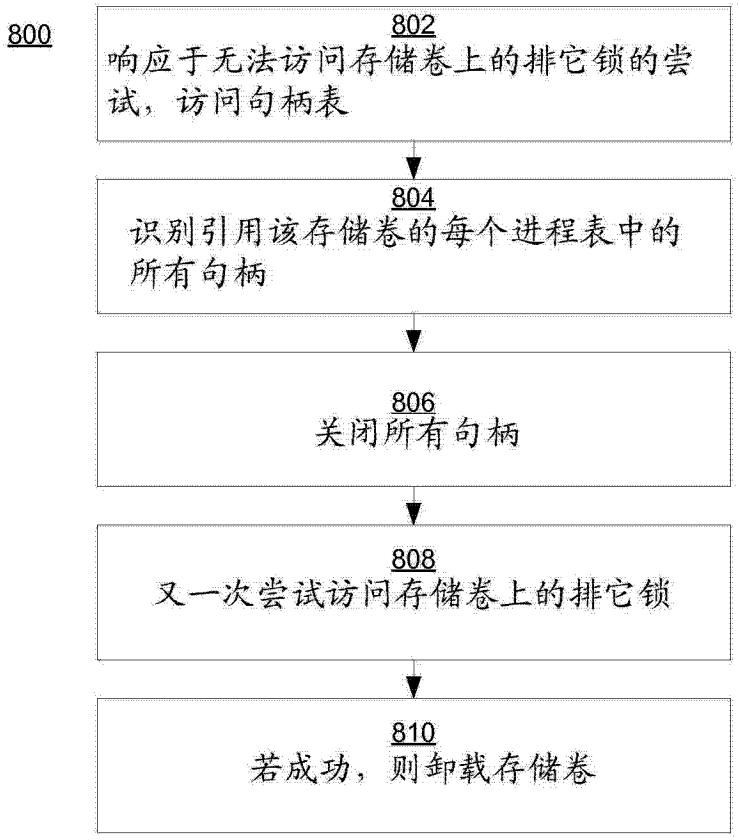

800
802
响应于无法访问存储卷上的排它锁的尝试，访问句柄表
804
识别引用该存储卷的每个进程表中的所有句柄
806
关闭所有句柄
808
又一次尝试访问存储卷上的排它锁
810
若成功，则卸载存储卷

图 8