



[12] 发明专利说明书

[21] ZL 专利号 95102127.3

[45] 授权公告日 2003 年 12 月 24 日

[11] 授权公告号 CN 1132109C

[22] 申请日 1995.2.24 [21] 申请号 95102127.3

[30] 优先权

[32] 1994. 3. 7 [33] US [31] 206706

[71] 专利权人 国际商业机器公司

地址 美国纽约

[72] 发明人 T·G·康特雷尔 S·扎吉

A·A·沙希恩 R·B·沃德

审查员 洪 岩

[74] 专利代理机构 中国专利代理(香港)有限公司

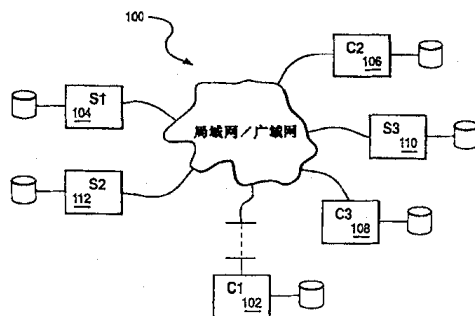
代理人 董江雄 马铁良

权利要求书 3 页 说明书 14 页 附图 6 页

[54] 发明名称 在一个分布式文件系统中高效高速缓存的系统与方法

[57] 摘要

一种高速缓存管理程序高效地支持连通的与断开的客户操作。为文件系统数据、状态及路径名信息维护一个单一的整体高速缓存器。可能时,文件系统请求由高速缓存器满足,而必要时则传送给一个远程文件系统去服务。高速缓存管理程序是与操作系统文件语法及分布式文件系统协议无关的。断开的操作是通过一种主动高速缓存策略与登录文件系统修改支持的。修改登录是在断开客户时通过登录目标而建立的,并在建立时对登录进行优化。



1. 一种管理在一个第一操作系统下操作的一个客户计算机系统中的一个文件系统高速缓存器的方法，包括下述步骤：
截取操作系统对一个分布式文件系统中的文件系统目标的请求；
5 求；
变换所述请求以消除依赖操作系统的语法；
根据所述变换后的请求，在所述客户计算机系统的存储装置中的一个高速缓存器中检测文件系统目标数据的存在；
如果高速缓存数据存在，便满足所述文件系统目标请求；
10 检测判定是否存在对所述分布式文件系统的连通；
如果不存在高速缓存数据但存在着连通，则不考虑分布式文件系统协议，生成一个对来自所述分布式文件系统的所述文件系统目标的请求；
将所述请求传输给所述分布式文件系统；以及
15 如果既不存在高速缓存数据又不存在连通，则拒绝所述文件系统目标请求。
2. 权利要求1的方法，其特征在于传输所述请求的步骤包括：
为所述请求的文件系统目标确定一种分布式文件系统协议；
将所述操作系统无关的请求变换成符合所述确定的分布式文件系
20 统协议的一个请求；以及
将所述变换后的请求发送给所述分布式文件系统。
3. 权利要求1的方法，其特征在于还包括下述步骤：
每当所述检测步骤判定不存在对所述分布式文件系统的连通时，
便执行下述步骤：
25 检测所述文件系统目标以确定一种请求类型；
为所述请求类型，激活一个请求登录目标；
调用一种修改登录方法来登录所述文件系统目标请求；以及
通过执行所述请求类型的登录优化处理，登录所述文件系统目标
请求。
- 30 4. 一种计算机文件系统，包括：
用于在一个存储装置中存储文件系统目标数据的高速缓存装置；
利用所述高速缓存装置来满足操作系统对文件系统目标的请求的

高速缓存管理装置，所述高速缓存管理装置是与操作系统语法无关的，并且与分布式文件系统协议无关；

5 截获装置，用于截获来自一个操作系统的对分布式文件系统的第一个第二操作系统控制下的远程服务器计算机系统中存储的文件系统目标的请求；

用于变换所述请求以消除依赖操作系统的语法的装置；

用于根据所述变换后的请求，在所述客户计算机系统的存储装置中的高速缓存器中检测文件系统目标数据的存在的装置；

10 用于如果高速缓存数据存在则满足所述文件系统目标请求的装置；

可拆卸地将所述高速缓存管理装置连接在一个文件系统目标服务器上的网络装置；以及

通过设置的网络装置存取文件系统目标请求的远程文件存取装置。

15 5. 权利要求4的系统，其特征在于还包括：

目标修改登录装置，用于在每当所述高速缓存装置从所述文件系统目标服务器上断开时，采集经所述高速缓存管理装置处理过的各操作系统请求的改变信息。

分解所述与语法无关的名字；

20 在所述客户系统中高速缓存分解后的名字，以及所述分解后的名字的名字路径成分的所有紧接子代的名字。

6. 一种用于维护文件系统信息的计算机系统高速缓存装置，包括：

用于存储多个文件系统目标数据项的存储装置；

25 用于将各所述文件系统目标数据项链接到一个文件卷的链接装置；以及

30 第二链接装置，用于将各所述文件系统数据目标链接到本地数据存储器的一个单元、本地存储器中的一个目录高速缓存器或本地存储器中的一个符号链路三者之一上；其中所述文件系统目标数据项是与任何文件系统目标名无关的，并且是可受各具有不同命名语法的多个文件系统目标父代引用的。

7. 权利要求6的计算机系统，其特征在于还包括：

用于每当从所述文件卷上断开所述高速缓存器时，为各文件卷登录对受所述多个文件系统目标数据项引用的目标作出的所有改变的登录装置。

8. 权利要求6的计算机系统，其特征在于还包括：

5 接口装置，用于接收操作系统对文件系统目标数据的请求，及将这些请求变换成与操作系统无关的请求。

9. 权利要求6的计算机系统，其特征在于还包括：

10 登录装置，用于每当从所述文件卷上断开所述高速缓存器时，为各文件卷登录对受所述多个文件系统目标数据项引用的目标作出的所有改变。

10. 权利要求8的计算机系统，其特征在于所述目录装置包括：

用于存储请求的文件系统数据目标的分解后的目录项的装置；以及

15 用于存储层次上紧接在所述请求的文件系统数据目标的各路径成分下方的目标的分解后的目录项的装置。

在一个分布式文件系统中高效高速缓存的系统与方法

本发明涉及管理数据的信息处理系统。更具体地，本发明涉及一个分布式文件系统中的客户的数据管理。更具体地，本发明涉及可支持连通或断开的客户操作的分布式文件系统客户中的高速缓存与登录。

计算机工作站的功能与存储容量正在提高。工作站原来是为单个用户执行一个或多个独立的任务用的。一个单位中的许多用户对工作站的大量使用促成了对工作站间的通信与用户之间的数据共享的需求。这导致了诸如图1中所示的分布式文件系统结构的开发。

图1中的若干工作站是用一个局域网(LAN)互联的。一个广域网(WAN)可用于互联若干局域网。LAN与WAN构成一个单一的逻辑网络。图1中的工作站标识为服务器S1、S2、S3及客户C1、C2、C3。将一个工作站指定为一个客户或服务器取决于该特定的工作站在网络中所执行的功能。一个特殊的工作站可以同时既是一个客户又是一个服务器。一个分布式文件系统的实现要求存在至少一个服务器及至少一个客户。

传统上，客户工作站在所有时间上都是连接在LAN上以接受该LAN的服务及共享文件等资源的。从LAN断开意味着从LAN资源上断开。只有在断开之前已将必要的文件与程序复制到客户上，工作才能继续在卸下的客户上进行。

便携式工作站的功能的提高已使分离的或“移动的”计算成为现实。便携式工作站可以与桌面或桌旁单元相同的处理器、存储器及盘存储容量购置。然而，便携式工作站或便携式计算机当前在分离时具有相同的限制：除非在断开之前已对文件或程序进行了复制，否则便没有共享的资源可利用。当重新连通工作站时，留给用户人工地使在便携式设备上改变过的文件与网络上文件一致。图 1 中示出了作为用虚线连接在网络上的客户 C 1 1 0 2 的一个可断开的便携式工作站。

诸如图 1 中所示的通过网络的数据文件共享已出现了多时。最简单的共享形式允许客户请求来自服务器上的一个文件的数据。将必要的的数据送至客户处理器并将对该数据的任何改变或修正返回给服务器。建立了适当的封锁使一个第二客户不改变被第一客户掌握的一个文件中的数据。

分布式文件系统通过增加更有效地将数据分配给客户与更有效地控制文件共享机制而改进文件共享。存在着许多分布式文件系统。一种流行的分布式文件系统便是 Transarc 公司推出的 Andrew 文件系统 (AFS)。

AFS 通过在正在存取服务器数据的一个客户中建立一个文件高速缓冲存储器而改进分布式文件处理的效率。这一高速缓存器是由客户应用程序访问的，并且只在高速缓存缺失时才从服务器取数据。数据的高速缓存减少了网络通信量并加速客户处的响应时间。

AFS 高速缓存器的兼容性是基于一个服务器回叫系统的。服务器会通知高速缓存了某些数据的各客户任何使客户的高速缓存数据无效的条件。一旦无效，便从高速缓存器中删掉这些数据，而在需要时必须重新从服务器得到。

连接在AFS 上的一个工作站的断开会使客户丧失分布式文件的存取。断开时，客户会自动失掉所有回叫而使高速缓存器无效。

分布式文件系统（DFS）是AFS 的改进产品，它支持OSF 分布式计算环境（DSE）。DFS 采用以服务器为基础的令牌机构来保证客户高速缓存器的兼容性。客户取得“读”令牌来保证它们的高速缓存器中的数据的有效性。如果一个客户要改变一个文件中的数据，它取得该数据的一个“写”令牌，赋予一个客户（诸如C1）一个“写”令牌使其它所有客户中的相同数据的“读”令牌失效。令牌失效使这些客户中的高速缓存数据失效。从一个DFS 系统上断开使客户丧失任何未完成的令牌。在没有令牌的情况下，客户不能读或写甚至高速缓存数据。

Carnegie Mellon 大学已作出努力来为连接在AFS 网上的可卸下的工作站提供支持。CODA 项目是旨在提供一个分布式文件系统以经常的数据可利用性的。它通过在多个服务器中复制数据及提供用户的断开操作支持来达到这一目的。见：“Coda：一种分布式工作站环境的高利用率文件系统” IEEE 学报，卷39，4号1990年4月。

Coda 中的分布式操作是通过数据的有利复制控制完成的。有利的控制允许许多客户读与写数据，即使在断开时也一样。数据不兼容性留待以后标识与解决。不利的复制控制通过限制在一个单一的部分上的读与写而防止所有的冲突。断开从该客户上去掉判定另一个客户是否具有对数据存取的能力。在不利的复制控制下，客户不能写。见

“Coda 文件系统中的断开的操作”，J. Kistler, M

Satyanarayanan, Carnegie Mellon 大学，第13次操作系统原理讨论会会议文集，1991年10月。

Coda 分布式操作允许客户继续存取与更新保持在客户侧高速缓存器中的数据。文件操作只在高速缓冲存储器出现缺失时才失败，因为这时客户才不能从服务器中存取数据。AFS 与Coda 高速缓存整个文件。Coda 在断开的操作期间更新文件并维护对数据的所有改变的登录。重新连通时，客户通过将登录的事务处理提交给服务器文件，负责更新数据的所有服务器复制品。数据不兼容性是通过推迟人工干预的决定而处理的。

Coda 具有一个主要的缺点，只支持Coda 服务器上的连接。Coda 与AFS 一样是一个分布式文件系统协议。Coda 依赖于服务器与所有客户都在使用相同的分布式文件系统协议这一事实。这便要求网络中的所有客户与服务器改变到Coda 协议来支持断开的操作。

Huston 与Honeyman 在“AFS 的断开的操作”，L. B. Huston 与P. Honeyman，信息技术综合中心，Michigan 大学，中提出了第二种方法，公布在USENIX 移动与位置无关的计算讨论会会刊，1993 年 8 月。Huston 与Honeyman 提出了一种能够不改变服务器而连接到一个标准的AFS 服务器的客户系统。客户代码修改支持断开的操作与重新连通。重新连通与一致性是通过登录断开时的每一个事务处理而完成的。每一次读、写或更新是记录在一个在重新连通时发送给服务器的一个事务处理登录中的。这一方法具有支持一个标准的AFS 服务器的优点。然而，它依赖于AFS 分布式文件系统协议的存在，并且不工作在AFS 服务器以外的服务器上。总的登录方法还降低系统效率。登录的大小还会在任何大小的系统中导致客户盘空间问题与一致性延迟。

从而，在提供一种允许断开的客户继续操作并在重新连通时具有高效一致性的分布式文件系统中存在着技术问题。另一个技术问题是

建立一个可使用多种分布式文件系统协议的分布式文件系统客户。最终，存在着建立一个独立于目标或路径名分解的操作系统语法的分布式文件系统客户的技术问题。

本发明旨在提供一种支持连通与断开的客户操作两者的分布式文件系统。该客户文件系统可与多种文件系统结构连接并支持高效的事务处理登录。

本发明旨在管理在一个第一操作系统下操作的客户计算机系统中的一个文件系统高速缓存器的一种方法。该方法包括下述步骤：截取对一个分布式文件系统中的文件系统目标的操作系统请求；交换所述请求以消除操作系统依赖的语法；在所述客户的存储装置中的一个高速缓存中测试有关所述文件系统目标的数据的存在；以及，如果高速缓存数据存在，便满足所述文件系统目标请求。

因此，本发明的一个目的为提供一种能够在从服务器断开时操作的分布式文件系统客户。

另一个目的为提供一种为重新连通时的一致性而高效地登录事务处理的一种可断开的客户。

本发明的又一个目的为提供给一个文件系统客户以高效的高速缓存以降低高速缓存所需的存储器与盘空间的量及提高高速缓存处理的速度与效率。

又一个目的为提供一种能以多种分布式文件系统协议工作的分布式文件系统客户。

本发明的又一个目的为提供一种独立于操作系统的分布式文件系统客户。

本发明的又一个目的为通过主动高速缓存分解的目标名而提高

高速缓存效率。

本发明的又一个目的为提供一种不依赖于一个目标名而允许各文件系统目标具有各有不同的目标名语法规则的多个父代的高速缓存管理系统。

从下述附图中所展示的本发明的一个较佳实施例的更具体的描述中，本发明的上述及其它目的、特征与优点将是显而易见的，附图中相同的参考号表示本发明的相同部分。

图 1 表示本发明的一个分布式网络环境。

图 2 为包含本发明的一个实施例的一个工作站的方框图。

图 3 为一个先有技术分布式文件系统的方框图。

图 4 为例示本发明的实施例的功能部件的方框图。

图 5 为本发明的较佳实施例中的数据结构的示例。

图 6 为本发明的一个替代实施例的方框图。

图 7 为描述本发明的一个实施例的处理步骤的流程图。

图 2 例示一种典型的客户或服务器工作站配置。本发明最好用诸如 IBM RISC (精简指令集计算机) 系统 / 6000 工作站或 IBM PS / 2 个人计算机等工作站实现。客户与服务器工作站用一个令牌环或以太网 LAN 互联，如图 1 中所示。连接在 LAN 上的工作站中可包含诸如 C 1 的便携式工作站，它可从 LAN 上断开，并独立于 LAN 使用。可以理解，在本发明的范围内可使用许多其它配置的工作站硬件或 LAN 类型。

一个典型的客户或服务器工作站 2 0 0 具有一个处理器 2 1 0、系统存储器 2 1 4 以及诸如硬盘驱动器、软盘或光盘存储设备等非易失性存储器 2 1 2。处理器通过 I / O 控制器 2 1 8 接受从诸如键盘 2 2 4 与指点设备 2 2 6 等输入 / 输出 (I / O) 设备的输入。系统

在图形控制器 2 2 0 驱动的显示器 2 2 2 上将图形信息提供给操作员。该工作站是通过网络接口适配器 2 1 6 连接到一个网络（未示出）上的。

本发明的较佳实施例是用操作在工作站 2 0 0 的存储器与处理器中的一个计算机进程实现的。实现本发明的一种计算机程序产品可存储在非易失性存储器 2 1 2 中，其中包括磁带、软盘、或激光盘只读存储器（CD-ROM）设备上的存储器。

参照图 3 例示了先有技术分布式文件系统的操作。图 3 例示了用一个网络接口 3 0 6 互联的一个客户工作站 3 0 2 与一个服务器工作站 3 0 4。用户进程 3 0 8 提出的对一个特定文件或有关位于盘存储器 3 9 0 上的一个文件的信息的请求的处理如下。注意，客户 3 0 2 与服务器 3 0 4 两者都分成用户地址空间 3 1 0、3 1 4 与核心地址空间 3 1 2、3 1 6。应用程序与核心之间的接口（称作应用程序接口或 API）为 S. R. Kleiman 的“Vnodes：在 Sun UNIX® 中的一种多文件系统类型的结构”，夏季 Usenix 会议会报，1 9 8 6，中所定义的 VFS/vnode API。当然也可采用其它接口。VFS/vnode 接口允许存在多个虚拟文件系统。分布式文件系统成为对客户工作站的一个虚拟文件系统。

用户进程 3 0 8 请求存储设备 3 9 0 上的一个数据文件。请求 3 2 0 被 VFS 接口截获并传送到常驻在客户核心地址空间 3 1 2 中的分布式文件系统客户 3 2 2。该分布式文件系统客户管理在网络上与服务器的通信。在网络 3 0 6 上的一则报文 3 2 6 中将该请求传送给服务器核心 3 1 6 中的分布式文件系统服务器 3 2 4。分布式文件系统服务器 3 2 4 通过使用 VFS 接口 3 2 8 存取存储设备 3 9 0，以存

取包含设备 3 9 0 的服务器的本地物理文件系统。得到所请求的数据，并将其在 3 3 0 处从分布式文件系统服务器 3 2 4 传送到分布式文件系统客户 3 2 2，后者又在 3 3 2 处将数据传送给请求的用户进程 3 0 8。

分布式文件系统客户 3 2 2 可包含一个数据高速缓存器（未示出）来高速缓存从服务器请求来的数据。分布式文件系统客户在向服务器请求数据之前，先在高速缓存中检验有效数据。

本发明的较佳实施例示出在图 4 中。客户工作站具有一个用户地址空间 4 0 2 及一个核心地址空间 4 0 4。本发明的改进的高速缓存管理程序 4 0 6 最好操作在用户地址空间 4 0 2 中，但也能实现在核心地址空间 4 0 4 中。

用户程序 4 0 8、4 1 0、4 1 2 发出文件系统请求 4 1 4、4 1 6、4 1 8。用户程序 n 示出为使用一个逻辑文件系统 4 1 4，后者依次发布一个文件系统请求 4 2 0。文件系统请求符合于 VFS/vnodes 接口，然而在本发明的范围内也能使用其它接口。本发明可用于任何符合 VFS/vnode API 的逻辑文件系统，或者在替代的实施例中，符合任何其它 API 的逻辑文件系统。

VFS 接口将文件系统请求导向高速缓存管理程序 4 0 6。高速缓存管理程序 4 0 6 维护有关客户所存取的文件系统的本地信息。如果可能，高速缓存管理程序服务于文件系统请求而不存取服务器。如果所请求的信息在高速缓存器中并且仍然有效，则文件系统请求可由高速缓存管理程序满足。下面将更详细地描述高速缓存管理程序 4 0 6。

高速缓存管理程序 4 0 6 从位于高速缓存管理程序存储器中的数

据中、或者通过存取本地物理文件系统 4 2 0、或者通过使用分布式文件系统存取 4 2 2 三者之一来满足文件系统请求。高速缓存管理程序在本地物理文件系统中保存分布式文件的本地拷贝供快速存取。分布式文件系统存取 4 2 2 向分布式文件系统服务器 3 2 4 发布必要的命令，如上面所讨论的。虽然本发明不限于整个文件的高速缓存，但较佳实施例是高速缓存整个文件的。

高速缓存管理程序 4 0 6 与分布式文件系统存取 4 2 2 独立于操作系统与分布式文件系统存取协议。操作系统独立性是通过支持定义的接口达到的。支持的接口包括：用于文件系统操作的VFS+ /Vnodes接口；诸如由IBM SOMobjects 持久性框架提供的持久性接口；用于高速缓存的文件的本地文件系统接口（LSI）；各种分布式文件系统协议；高速缓存同步接口；以及登录重放接口。高速缓存管理程序不依赖于任何操作系统语法。文件名是变换成独立于文件名成分分隔符字符的，并认为是没有保留字符与保留字的。路径名分解也是独立于操作系统语法的。从而，一个特定的文件系统目标可由具有冲突的语法规则的两个操作系统引用。例如，高速缓存管理程序 4 0 6 可实现为与IBM OS/2® 操作系统或IBM AIX® 操作系统一起操作，或只加以很小的改动便能与其它任何操作系统一起操作。

本发明的高速缓存管理程序是独立于分布式文件系统协议的。分布式文件系统存取 4 2 2 可使用来自开放软件基础（Open Software Foundator）（OSF®）的分布式计算环境（DCE®）服务器或IBM LAN 服务器程序产品两者之一来存取远程服务器。只需小的改动便能使用其它远程文件存取协议。同步机构与登录重放机构操作与协议相关的任务。

高速缓存管理程序 4 0 6 维护客户高速缓存器 4 0 9。结合高速缓存器 4 0 9，高速缓存管理程序维护卷信息 4 1 0 与一个修改登录 4 1 2。卷信息 4 1 0 中包含关于客户工作站能通过高速缓存管理程序存取的文件系统的信息。在本最佳实施例中，远程文件系统是按卷连通到客户上的。各卷在卷信息数据库中有一个卷项。因此，一个远程文件系统将作为一个 Unix 型文件系统（例如作为一个特殊文件系统 / rfs 安装的），或者作为一个 DCE 驱动器字母（例如 G : \.）连通到客户工作站上。关于远程文件系统及其状态的所有信息维护在卷信息数据库 4 1 0 中。多个远程文件系统可同时连通到一个单一的客户上。

图 5 中示出了高速缓存器 4 0 9 的详细结构。高速缓存器 4 0 9 中为文件路径名、文件系统目标数据与文件系统目标状态信息维护一个单一的高速缓存器。高速缓存器是维护在永久性堆中的，即周期性地写入非易失性存储器中，并且在电源或系统故障时能从其中恢复。一个单一的整体高速缓存器减少总的高速缓冲存储器需求，从而提高高速缓存器性能。单一高速缓存器只需要一个散列表而不是先有技术系统中的三个。所有的高速缓存处理可实现在单一的一组代码中，它存储不同类型的文件系统目标，而不是用于各种类型的高速缓存器的分离的代码的传统机制。高速缓存的文件系统目标一般性地示出在 5 0 2 处。文件系统目标 5 0 2 既用于高速缓存管理程序的连通操作，也用于其断开的操作。

文件系统目标 5 0 2 是一种包含存取一个远程文件或该远程文件的本地高速缓存的拷贝的必要信息的数据结构。为客户工作站引用的各远程文件或目录在高速缓存器中建立一个文件系统目标项。文件系

统目标中包含一个VFS+ / vnode 标识符 5 0 4 与文件标识符 5 0 6。文件标识符 5 0 6 中包含卷、服务器与文件数据。VFS+ / vnode 标识符中包含VFS / vnodes 标准规范所定义的信息。包括进这些信息允许对所有VFS / vnode 接口的全面支持。本发明在该数据结构上附加额外的信息来支持改进的高速缓存与断开的操作。卷项 5 0 8 为对卷信息数据库 4 1 0 中该文件所属的卷的卷项 5 4 5 的一个指针。维护了一个状态变量 5 1 0 与状态结构 5 1 2 来为数据提供高速缓存状态信息。在使用诸如 C++ 程序设计语言等工具实现一个面向目标的环境中的本较佳实施例中，状态结构是定义为一个目标类的。一个特定的操作系统文件（诸如OS / 2 或AIX）的状态结构以建立状态结构类的一个子类定义。该子类用定义文件状态所必需的实际内容超越状态结构类。

文件系统目标维护高速缓存器相关性信息 5 1 4。这一信息用于保证高速缓存器与服务器及服务器复制品之间的一致性。存取控制清单（ACL）的一张清单 5 1 6 按照IEEE POSIX® 标准定义文件系统存取与修改的许可。其它标志信息维护在 5 1 8 处。卷的安装状态及父代指针 5 2 2 与优先级信息 5 2 4 维护在 5 2 0 处。各文件系统目标具有一个分配的优先级，它用于确定删除哪些目标来为新目标让出空间。

文件系统数据是通过一个文件 / 目录 / 符号链路指针 5 2 6 存取的。这一指针取决于文件系统目标的类型，提供对三种不同类型的高速缓存数据的存取。一个数据文件的文件系统目标有一个指向高速缓存文件 5 4 0 的指针，高速缓存文件 5 4 0 又指向本地文件系统 420 中的本地高速缓存的数据文件。一个目录指针指向一个高速缓存的目

录 5 4 2。该高速缓存的目录 5 4 2 是建立在一种主动高速缓存方式上的。每当请求一个特定目录的目录信息时，便返回该目录及其全部紧接子代。由于在请求了父代目录信息之后，立即请求关于一个或多个子代目录的信息的概率很高，因此这能降低高速缓存缺失。目录高速缓存器提供路径名分解所需的信息，即将一个特定的文件系统路径请求映射到一个特定的数据文件上。最终，指针 5 2 6 可指向高速缓存的符号链路信息 5 4 4。这一高速缓存的信息允许符号链路的分解。

在 5 2 8 为各种计数器、门锁与柄提供了项，以及一个清除状态指示器 5 3 2。对于断开的操作，清除状态指示器 5 3 2 是重要的，并用于指示一致性状态。

注意到文件系统目标名并不包括在数据结构 5 0 2 中是重要的。这便允许从不同的目标名设定无冲突地引用该文件系统目标。各文件系统目标能有多个父代。各父代可遵守不同的命名约定，例如目标管理群 (OMG) 或 AIX、OS/2 或其它操作系统的命名规定。从而，一个特定的文件目标 `employee-list` 从 OS/2 可作为 `C:/`

`employee-list` 及从 AIX 作为 `/usr/employee-list` 来存取。

高速缓存管理程序 4 0 6 支持连通的与断开的操作。在连通的操作中，高速缓存管理程序 4 0 6 从高速缓存器或者必要时通过远程文件存取来满足文件系统请求。在断开时，高速缓存管理程序从高速缓存器满足请求，并维护一个对所有文件系统修改的修改登录 5 4 6。

图 6 表示本发明的一种高速缓存管理程序。这一实施例是设计成用于 IBM OS/2 操作系统的。文件系统请求由系统作为一条“磁盘操作系统调用”命令 6 0 2 发布。所有的文件系统请求是由 IFS 文件系统 6 0 4 服务的。IFS 提供逻辑文件系统能力。IFS 可安装及服务于

不同类型的文件系统。安装的文件系统中可包括诸如DOS系统常用的FAT文件系统606、或者OS/2 HPFS文件系统608。在LAN环境中，LAN请求器610呈现为另一个文件系统，其中文件存取是传送给LAN服务器来操作的。按照本发明的一个高速缓存管理程序可作为另一类型的文件系统612安装。所有的远程文件系统存取是传送给高速缓存管理程序612的。高速缓存管理程序检测高速缓存器614（在用户地址空间中的），以判定所请求的信息是否已经高速缓存了。如果是，便将该信息提供给应用程序，包括在必要时提供或修改保持在本地文件系统606或608中的高速缓存的数据。如果未找到文件系统数据，便将一个请求传送给LAN请求器610，令其在LAN服务器上存取必要的的数据。当远程数据到达时，便将它放在高速缓存器中供将来引用。

图7表示一条文件写命令的处理流程的流程图。其它文件系统命令是类似地实现的。该命令从702开始，在704发布一条文件写命令。高速缓存管理程序在706检验判定所请求的文件是否在高速缓存器中。如果是，便在708更新该文件的高速缓存拷贝。接着，在710检验高速缓存器以判定该卷是否是当前连通的。连通状态是维护在卷信息数据库410中的。如果该卷是连通的，便在712将改变传播回服务器而处理在722结束。如果该卷不是连通的，便在714将改变登录在修改登录中而处理在722结束。

如果请求的文件不在高速缓存器中，接着高速缓存管理程序便在716检验所请求的卷是否是连通的。如果是连通的，便在718从远程文件系统请求该文件、将其放在高速缓存器中，并在步骤708恢复处理。如果该卷不是连通的，便在720将一个失败信号返回给

应用程序。

将修改（诸如步骤714）登录进修改登录546是通过登录目标完成的。问题域是由一个客户修改登录类及登录的各不同类型的事务处理的一个独立的类表示的。为各种操作定义了不同的目标，诸如对于写文件、建立文件、删除文件都存在一个目标。当在断开的操作模式中在714处高速缓存管理程序登录了一个改变时，便建立了一个与所请求的操作对应的登录目标类型（诸如写文件）。调用一种登录方法将目标放入登录中。该方法包括压缩与优化供重放的登录的方法。登录一个特定的目标可能会导致删除一个以前登录的目标。例如，当建立一个文件并稍后删除它时，文件删除目标将删除文件建立登录项、任何文件修改登录项、然后退出而不在登录中放置一个文件删除项。本系统保证重新连通时，高速缓存器与远程文件系统的同步会快捷高效地发生。登录目标中的信息消除了重放前处理或压缩登录的必要。

从以上描述中可以理解，在本发明的较佳实施例中可作出各种修正与改变而仍不脱离其真实精神。这一描述只是为了例示的目的，不应视为是一种限制的意思。本发明的范围只受以下权利要求书限制。

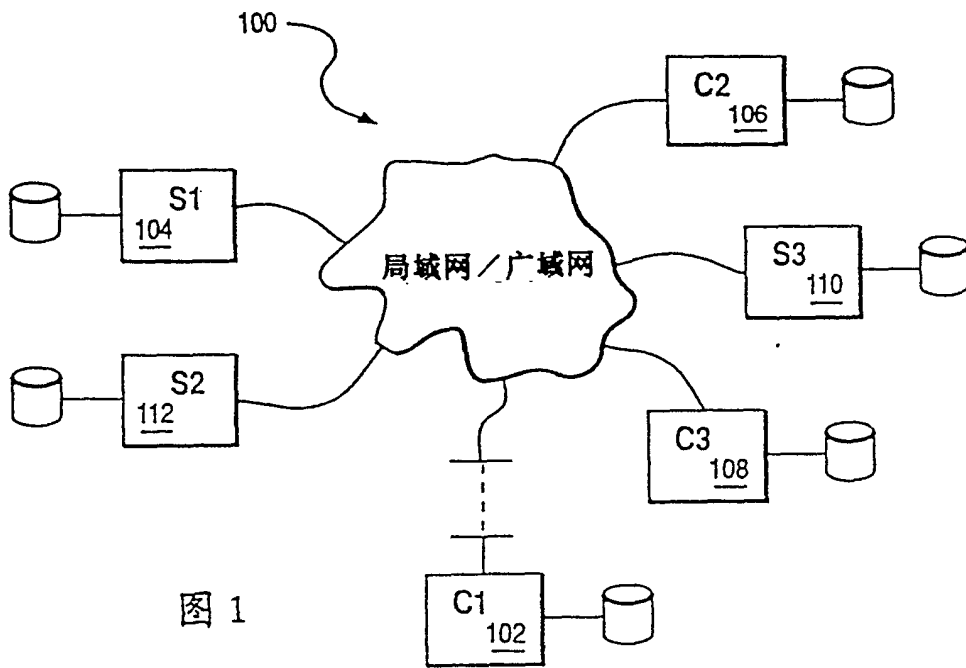


图 1

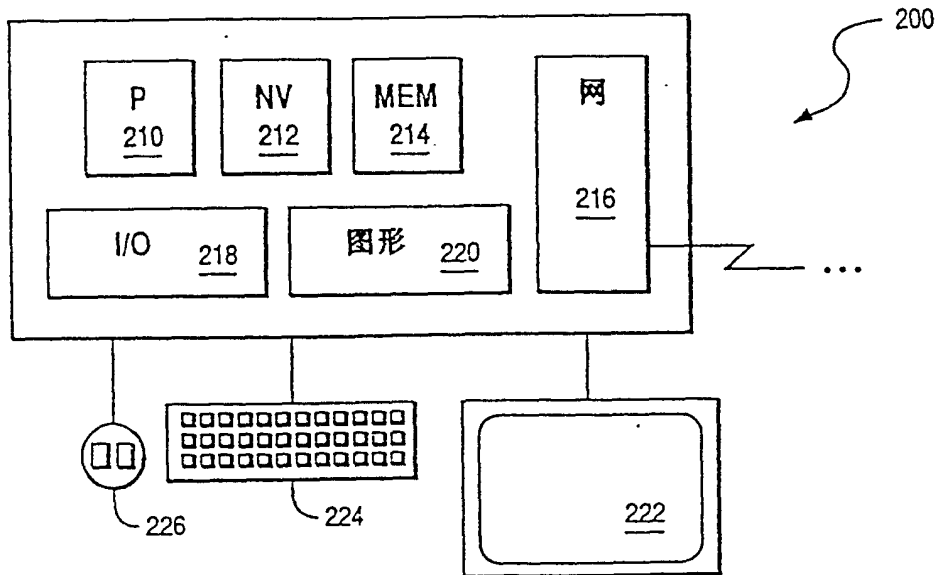


图 2

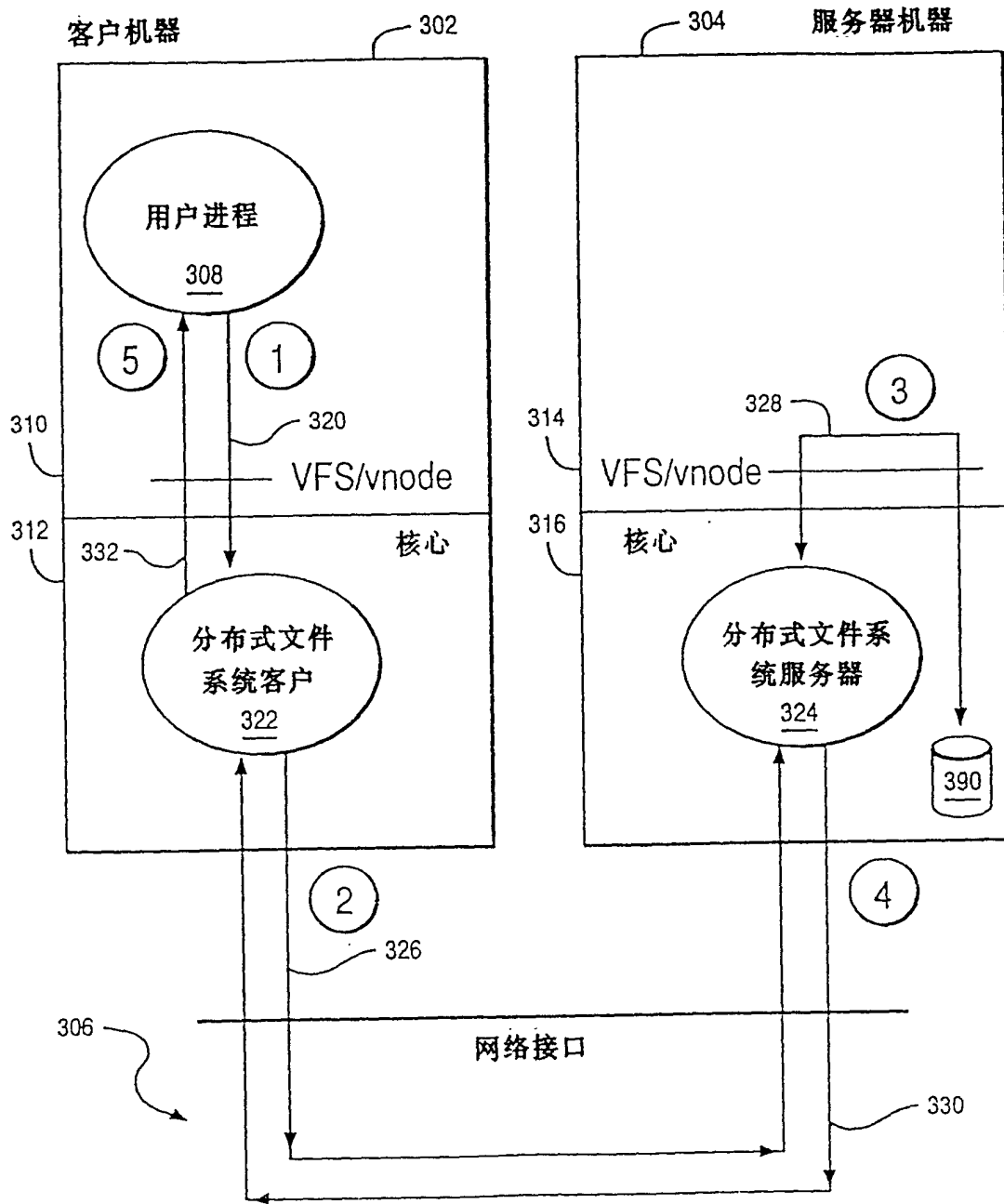


图 3

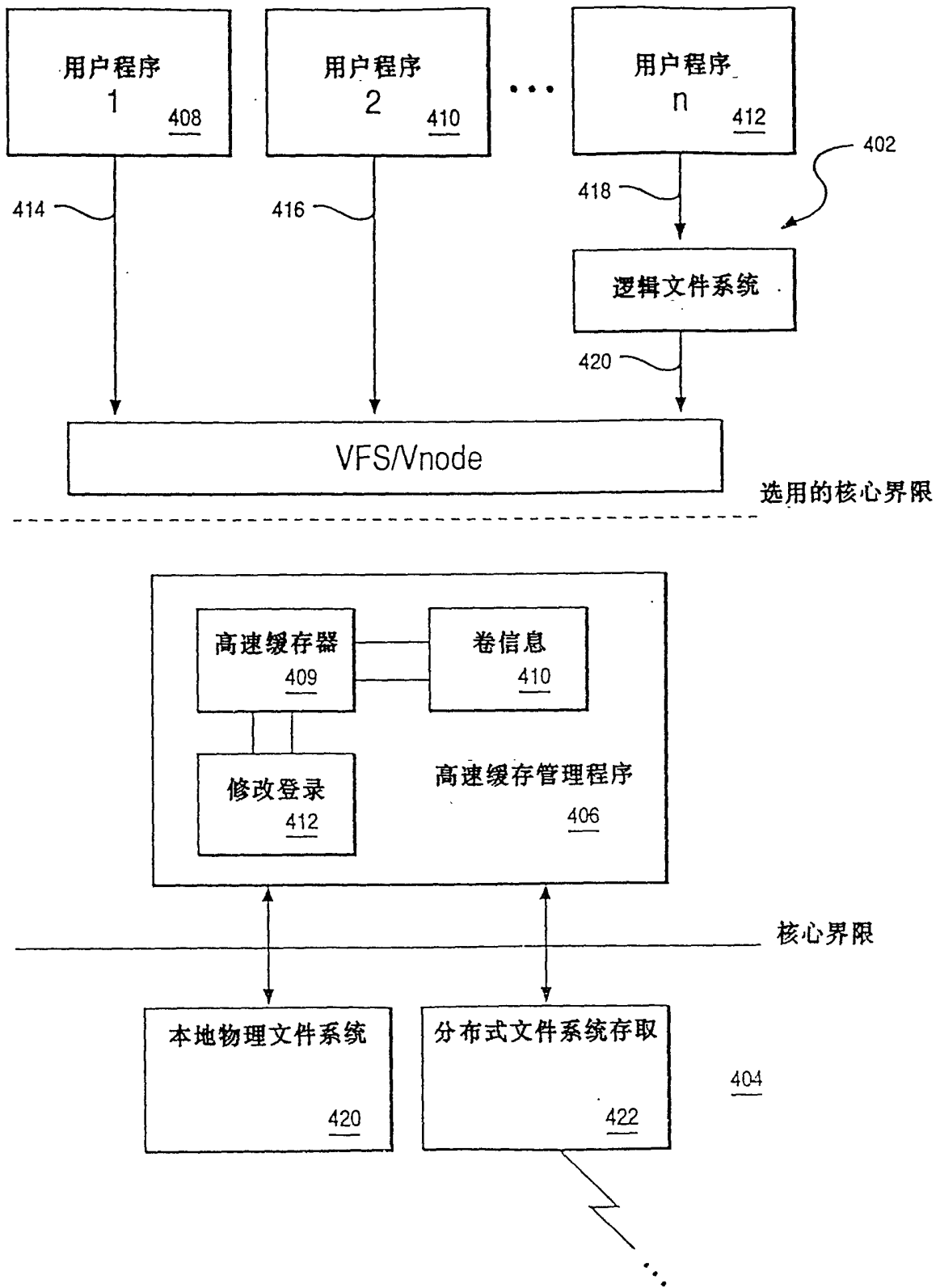


图 4

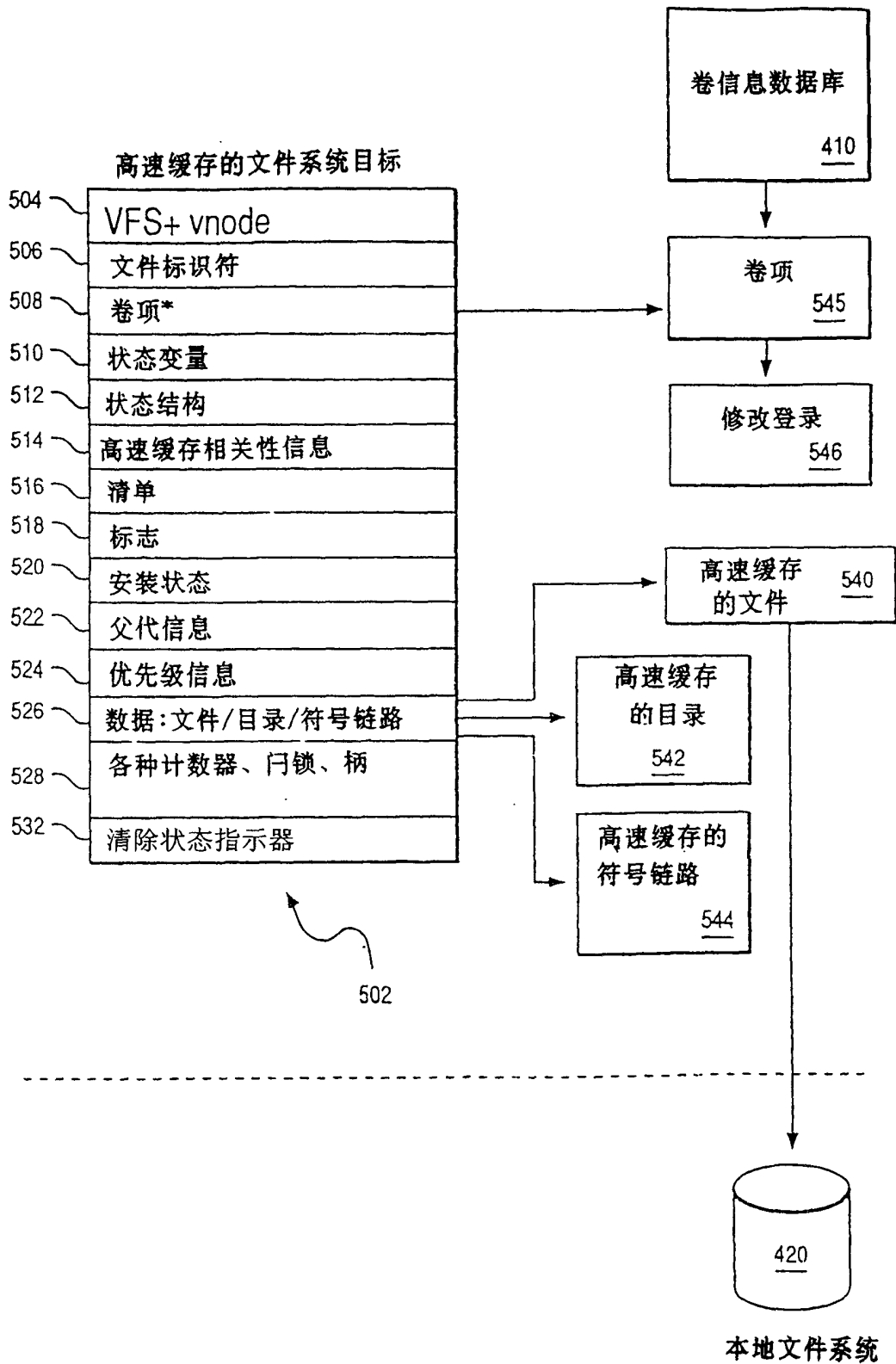


图 5

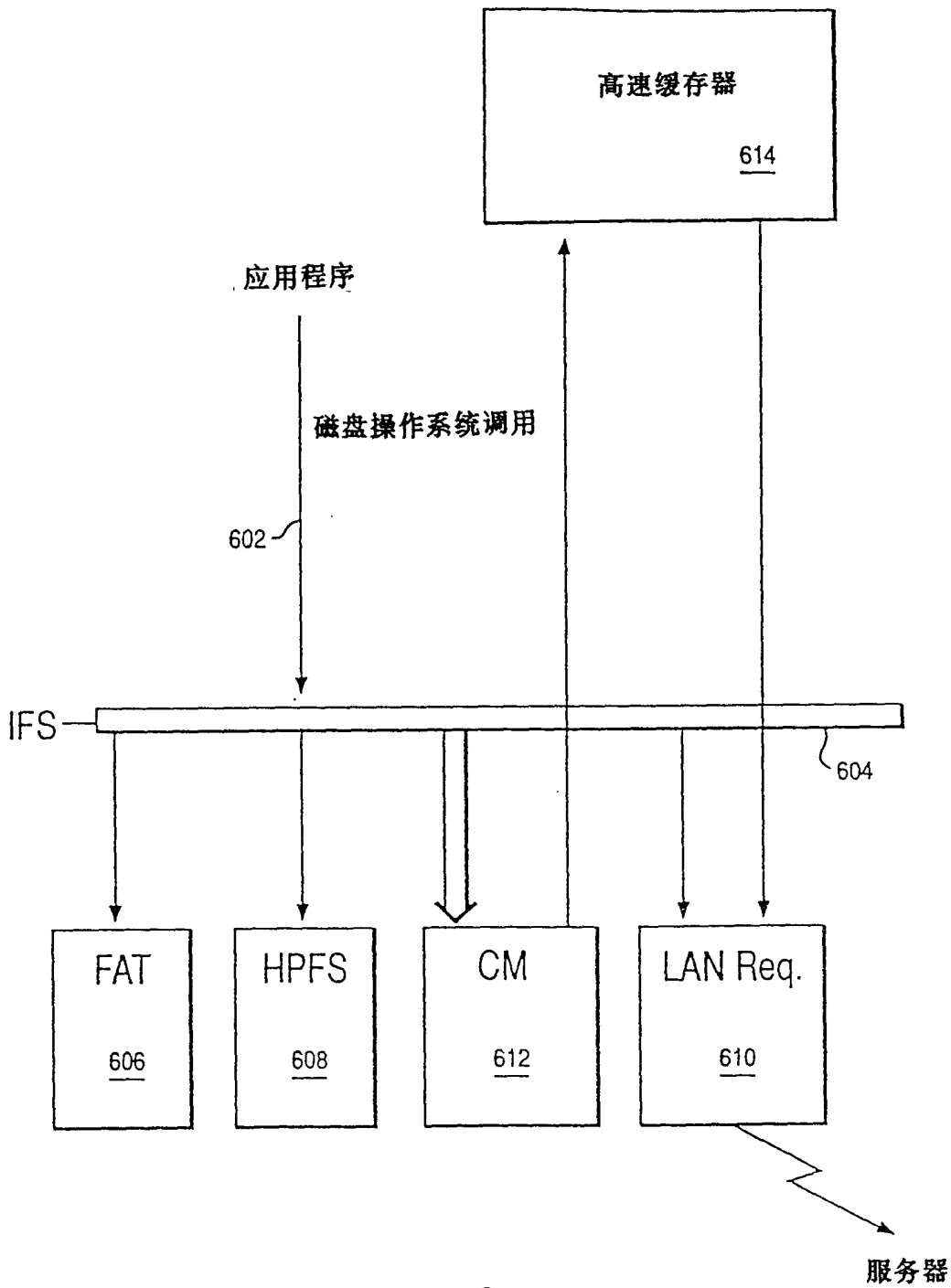


图 6

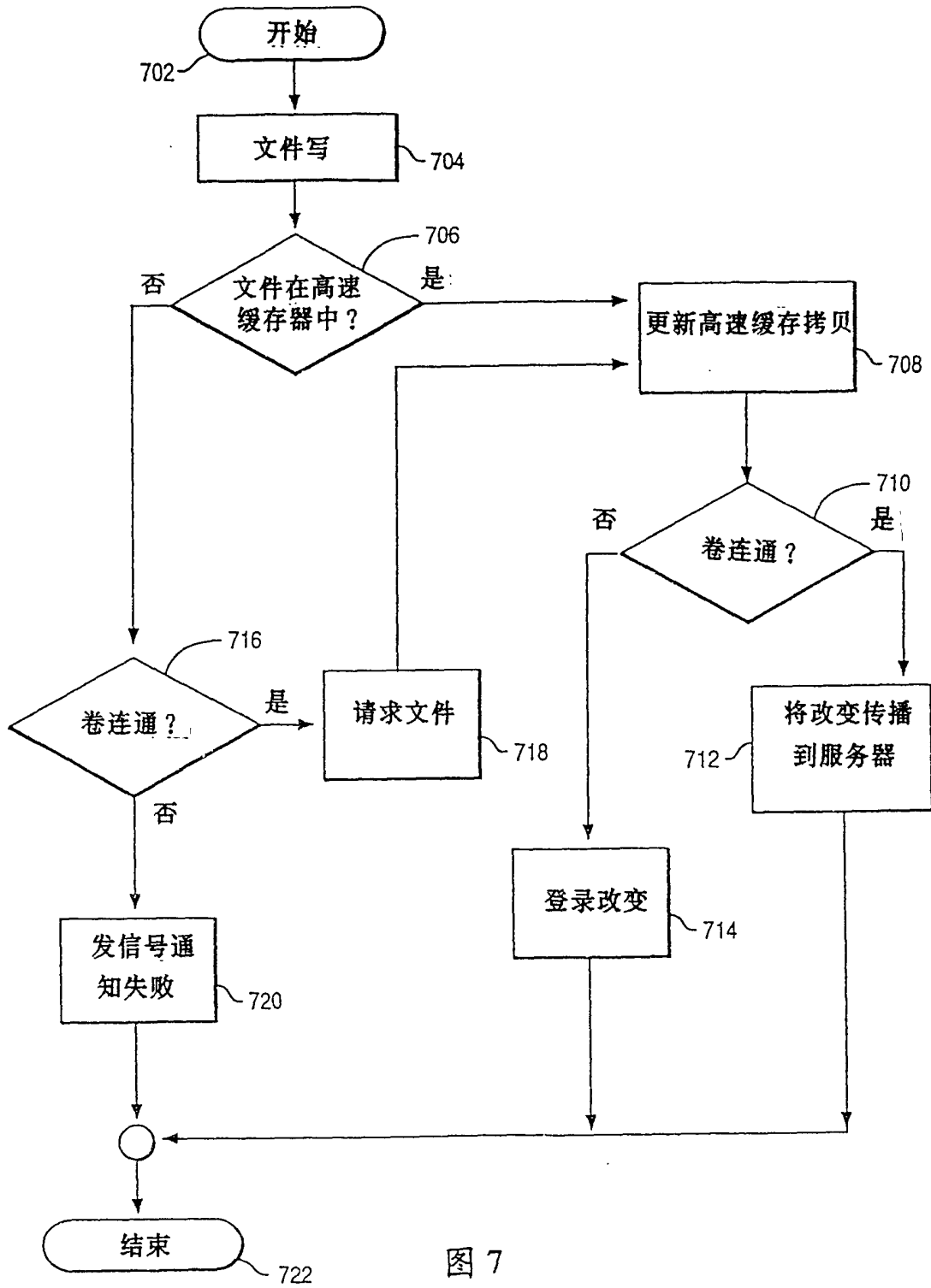


图 7