

[51] Int. Cl⁷

H04L 12/58

H04L 29/06



[12] 发 明 专 利 说 明 书

[21] ZL 专利号 00816711.7

[45] 授权公告日 2004 年 12 月 1 日

[11] 授权公告号 CN 1178436C

[22] 申请日 2000.11.3 [21] 申请号 00816711.7

[30] 优先权

[32] 1999.11. 4 [33] US [31] 09/434,048

[86] 国际申请 PCT/US2000/030500 2000.11.3

[87] 国际公布 WO2001/033798 英 2001.5.10

[85] 进入国家阶段日期 2002.6.4

[71] 专利权人 米雷波因特公司

地址 美国加利福尼亚州

[72] 发明人 萨蒂什·拉马钱德兰

布拉德利·A·泰勒

审查员 徐 刚

[74] 专利代理机构 北京市柳沈律师事务所

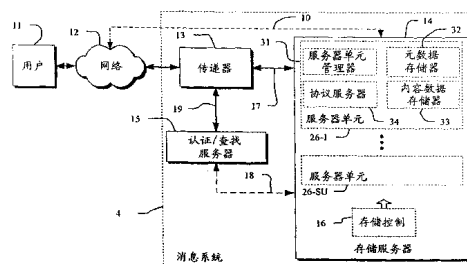
代理人 马莹 邵亚丽

权利要求书 2 页 说明书 22 页 附图 6 页

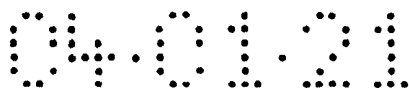
[54] 发明名称 电子消息传递系统

[57] 摘要

一种用于消息在网络上的通信的消息传递系统，其中消息包括元数据和内容数据。认证单元对来自网络和存储消息的存储服务器单元的消息对用户进行身份认证。存储服务器单元包括：多个协议服务器单元，用于依据网络上的消息所使用的协议处理消息；元数据存储单元，用于存储消息的元数据；内容数据存储单元，用于存储消息的内容数据；管理器单元，用于对元数据存储单元和内容数据存储单元进行公共控制。管理器单元包括公共寻址单元，用于对多个协议服务器单元的消息，对在存储服务器单元内位的置的消息的地址进行公共管理；和公共存取控制单元，用于控制多个协议服务器单元对存储服务器单元内的位置的存取。

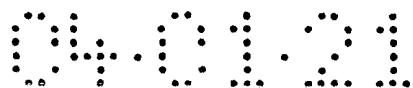


ISSN 1008-4274



权 利 要 求 书

1. 一种用于用户 (11) 经过网络 (12) 的消息通信的电子消息传递系统, 包括: 用于从和向网络 (12) 接收以及传递消息的装置 (13), 所述消息包括
5 元数据和内容数据; 用于存储消息的存储服务器装置 (14), 所述存储服务器装置 (14) 包括: 多个协议服务器单元 (26-1, ..., 26-SU), 用于依据网络 (12) 上的消息所使用的协议处理消息,
其特征不在于, 所述电子消息传递系统还包括:
用于对来自网络 (12) 的消息对用户进行身份认证的装置 (15),
10 所述多个协议服务器单元 (26-1, ..., 26-SU) 的每一个包括:
元数据存储装置 (32), 用于存储消息的元数据;
内容数据存储装置 (33), 物理上不同于所述元数据存储装置, 用于存储消息的内容数据;
管理器装置 (31), 用于对元数据存储装置和内容数据存储装置的公共控
15 制, 所述管理器装置包括:
公共寻址装置 (7-2), 用于对多个协议服务器单元 (26-1, ..., 26-SU) 的消息, 对在存储服务器装置 (14) 内位置的消息的地址进行公共管理;
公共存取控制装置 (7-1), 用于控制多个协议服务器单元 (26-1, ..., 26-SU) 对存储服务器装置 (14) 内所述位置的存取。
20 2. 如权利要求 1 的电子消息传递系统, 其中, 所述公共存取控制装置 (7-1) 包括公共锁定装置, 用于对于多个协议服务器单元 (26-1, ..., 26-SU)、对在存储服务器装置 (14) 内的所述位置上的锁定 (82) 的公共控制。
3. 如权利要求 2 的电子消息传递系统, 其中, 所述锁定 (82) 被每个所述存储服务器装置 (14) 存取的锁定算法 (92) 控制。
25 4. 如权利要求 3 的电子消息传递系统, 其中, 每当所述协议服务器单元 (26) 的连接被终止时, 所述锁定算法终止锁定。
5. 如权利要求 1 的电子消息传递系统, 其中, 经过网络 (12) 的消息使用的所述协议包括用于对用户进行消息传递的 SMTP。
6. 如权利要求 1 的电子消息传递系统, 其中, 经过网络 (12) 的消息使
30 用的所述协议包括用于由用户进行消息存取的 POP。
7. 如权利要求 1 的电子消息传递系统, 其中, 经过网络 (12) 的消息使



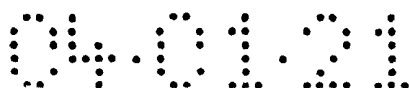
用的所述协议包括用于由用户进行消息存取的 IMAP。

8. 如权利要求 1 的电子消息传递系统, 其中所述公共存取控制装置(7-1)包括:

公共锁定装置, 用于控制在所述位置的锁定(82), 由此对于多个协议服务器单元(26-1, ..., 26-SU), 存储服务器装置(14)内的所述位置的锁定(82)在管理器装置(31)的公共控制之下。

9. 如权利要求 1 的电子消息传递系统, 其中所述公共存取控制装置(7-1)包括:

公共开放装置, 用于控制所述位置的开放, 由此对于多个协议服务器单元(26-1, ..., 26-SU), 存储服务器装置(14)内的所述位置在管理器装置(31)的公共控制之下。



说明书

电子消息传递系统

5 版权声明

这个公布专利文件的一部分包括受版权保护的素材。版权拥有者不反对任何人传真复制专利和商标局专利文件或记录中出现的专利文件或专利公布，但另外保留所有版权的无论什么权利。

10 技术领域

本发明涉及电子消息传递系统，并尤其涉及能够使用不同互联网协议的消息传递系统。

背景技术

15 在早期的电子邮件消息传递系统中，邮件消息典型地被传递到一个组织内的单个分时机器，并且每个用户登录到那台机器来读取用户的电子邮件。今天，电子邮件用户通常有一台或多台在办公室的机器、在家里的个人计算机、以及便携式计算机，于是需要一个适应不同操作模式的分布式消息传递体系结构。在分布式消息传递系统中，有三种一般的操作模式，即离线
20 (offline)、在线(online)、断线(disconnect)。当应用客户端/服务器体系结构时，客户端是例如一台工作站或个人计算机。

在离线操作中，消息通过网络传递到一台共享服务器，并且用户周期性地连接到此服务器并下载所有的未决消息到客户机。客户邮件从服务器取消息到运行邮件程序的客户机，并且消息被从服务器删除。因此，消息处理过
25 程是本地的，在客户机上。

在在线操作中，消息被留在邮件服务器上，并且被邮件客户程序远程操作——邮件客户程序在同一时间可能多于一个，并且在不同时间可能多于一个。

在断线操作中，邮件客户端连接到邮件服务器，对选择的消息制作“高速缓存”副本，然后从服务器断开连接，以后再与服务器重新连接并重新同步。用户于是可在离线的消息高速缓存上操作，但是这种模式是不同于离线
30

模式的，在于消息的最初副本保留在服务器上，并且邮件客户程序将随后重新连接到服务器，并在服务器与客户的消息高速缓存之间重新同步消息状态。在线和断线操作彼此互补并可在它们之间替换；然而，两者均不与离线操作兼容，因为根据定义，离线操作意味着在消息被复制到客户端的本地磁盘上以后，从服务器删除消息。

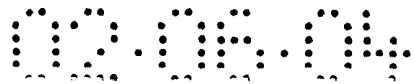
几个客户端-服务器协议的任何一个均可用于存取远程消息存储，包括通用文件存取协议以及特殊应用协议。每当邮件被传递到一台机器却在另一台机器上读取时，需要网络协议以在服务器机上存取消息。当使用不同机器时，必须作出判定使用哪个协议来存取消息数据。此问题将两者应用于输入消息文件夹（例如用户的 INBOX），也同样应用于保存消息文件夹。当读取输入消息文件夹时，通常的操作是保存消息到存档文件夹，于是两个数据设置（set）必须同时可用。选择的协议可以是普通远程文件系统存取协议（例如，NFS、SMB）、特殊应用消息存取协议，例如邮局协议（POP）以及互联网消息存取协议（IMAP）。

普通远程文件系统存取协议一般不是远程消息存储的电子邮件存取的选择，因为没有单个文件系统对所有计算机普遍地可用，安装和操作是困难的，并且经常导致网络带宽不够用。

特殊应用协议是远程消息存储的电子邮件存取的通常选择，因为这样的协议可被设计为最优化性能，能够在客户端与服务器之间对处理过程提供逻辑划分以最小化经过网络传输的数据，能够被无特殊优先权安装，并能够将客户端程序与服务器上使用的文件格式隔离。尽管可使用专有/厂家指定的解决程序和 X.400 P7 消息存取协议，互联网消息存取协议（POP、DMSP、以及 IMAP），特别是 POP 和 IMAP，是被广泛接受的一些协议。

邮局协议（POP）源于 1984 年并经过多次修改，现行的是 POP3。POP 被特别设计用于支持离线存取，但是有同样存在于其他两种范例的限制。分布式邮件系统协议（DMSP）在 1986 年初次定义并已被修改。不像 POP，DMSP 没有被广泛支持并大部分仅限于 PCMAIL 这个单独应用。DMSP 被特别设计以适应在 PCMAIL 中支持的断线操作。

互联网消息存取协议（IMAP）起源于 1986 年并经过多次修改，现行的版本是 IMAP4。IMAP 原来被设计用于支持在线存取模式。因为 IMAP 包括 POP 的功能扩展集能力，能够与 POP 一样完全支持离线存取，并且在版本 4 中所



作的附加，使它能够同样支持断线操作。最新的 IMAP 版本因此包括 POP 和 DMSP 两者的功能。

电子消息传递从电子邮件消息扩展到包括电子邮件、传真、声音邮件以及组件的任何电子消息形式。电子邮件的许多需要和限制扩展到电子消息传递的其他形式，并需要一个相干(coherent)并集成的电子消息传递系统，使用不同类型电子消息来操作。

在与消息传递系统的连接中，想要使用数据库作为信息的存储器。然而，传统数据库在消息传递系统环境中很难使用，因为消息传递系统需要以有效的方式升级，以容纳越来越大的消息量，而并不降低存取速度。

- 10 在美国的标题为“电子邮件信息的分配和存储系统(System For The Distribution And Storage Of Electronic Mail Information)”的专利 5940823 中，公开了电子邮件信息的分配和存储系统。此系统包括：一个服务器；驻留在服务器上的状态聚集(rendezvous)程序；以及一个包括驻留在服务器上的分配管理系统的分配存储设备；以及驻留在服务器上的外壳
- 15 (container) 管理系统。电子邮件信息的分配和存储方法包括以下步骤：
- (1) 从第一远程连接接收入站(inbound)信息，入站信息包括分配；(2) 处理分配，该处理导致分配被存储在存储器中；(3) 向存储介质写分配；(4) 从存储介质调页分配到存储器；(5) 将部分存储器中的分配向至少第二连接传递；以及(6) 响应于通报的接收者的存在，判定意向接收者。元数据和
- 20 内容数据被存储在单一文件中。

- 在 NADEAU M 的 “Your E-mail Is Obsolete” Byte, MCGRAW-HILL INC. ST PETERBOROUGH, US, vol. 22, no. 2, 1 February 1997, pages 66-69, 72, 74, XP000680521 ISSN: 0360-5280 中，描述了一个具有认证的电子邮件系统。
- 当发送消息时，使用 RSA 公共密钥方法在桌面对其加密。发送者用一个本地
- 25 或远程认证服务器检查得到接收者的公共密钥。在发送者的消息服务器上，消息被使用基于 RSA 的 S/MIME 协议再次被加密。同时，消息的复制去到发送者或第三方位的认证服务器。接收者的消息服务器将 S/MIME 加密进行解密，并用认证服务器检查以确保消息在传输过程中没被改变。最后，消息在桌面使用接收者的公共密钥被解码。

- 30 依照上述背景，本发明提供了一个相干并集成可升级的电子消息传递系统，能够用不同互联网协议有效操作，并能够操作不同形式的电子消息。



发明内容

本发明是一个消息传递系统，用于消息在网络上的通信，其中消息包括元数据和内容数据。认证单元对来自网络的消息对用户进行身份认证，以及
5 存储服务器单元存储消息。存储服务器单元包括：多个协议服务器单元，用于依据网络上的消息所使用的协议处理消息；包括元数据存储单元，用于存储消息的元数据；包括内容数据存储单元，用于存储消息的内容数据；并包括管理器单元，用于对元数据存储单元和内容数据存储单元的公共控制。该元数据和内容数据存储单元是物理上不同的单元，用于在不同的文件中存储
10 元数据和内容数据。管理器单元包括：公共寻址单元，用于对多个协议服务器单元的消息，对在存储服务器单元内位置的消息的地址进行公共管理；以及公共存取控制单元，用于控制多个协议服务器单元对存储服务器单元内位置的存取。

从以下结合附图的详细描述中，本发明的上述和其他目的、特性以及优点将会更加明显。

附图说明

图 1 描述了电子消息传递系统。

图 2 更详细描述了图 1 的消息传递系统。

20 图3描述了图2的存储服务器的一个实施例。

图 4 描述了图 2 的存储服务器的另一个实施例。

图 5 更详细描述了图 1 和图 2 的电子邮件系统。

图 6 更详细描述了图 1 和图 2 的电子邮件系统中的用户。

图 7 描述了服务器单元管理器的公共单元。

图 8 详细描述了服务单元管理器。

图 9 描述了电子邮件系统中一个邮件消息的表示。

5 图 10 描述了电子邮件系统中另一个邮件消息的表示。

具体实施方式

电子消息传递系统-图 1

10 图 1 描述了使得用户 11 能够向和从电子消息传递系统 4、并因此能向和从其他用户发送和接收电子消息电子消息传递系统。用户 11 是任何类型的位于人们办公地点、家中、或位于其他固定或移动位置的机器。用户 11 能够在人的控制下或脱离人的控制下操作。典型地，电子消息传递系统有适应包括离线（下载和删除）、在线和断线模式的不同操作模式的客户机/服务器分布式消息传递体系结构。发送和存储的消息是包括电子邮件、传真、声音以及组件的任何类型的电子消息。

20 图 1 的电子消息传递系统使用专用互联网协议，包括 POP、IMAP 以及 SMTP 或其他与经过网络 12 的互联网操作和通信兼容的协议。被用户用于经过网络 12 存取消息的 POP 和 IMAP 协议，都基于用于经过网络 12 发送消息的协议 SMTP。用户 11 可以是游牧的（nomadic）并与任何特定远程文件协议无关。典型地，用户 11 可以通过连接 17 发送、检索和保存消息，并可以管理存储服务器 14 内的远程用户文件夹。电子消息传递系统基于每条消息进行操作，来检索和更新状态信息。用户可以检索和更新个人设置信息并可以共享邮箱。电子消息传递系统对用户 11 的邮件传递，通常是传递到一个共享并一直可用的存储服务器 14，它允许从各种类型客户平台存取新的消息，25 并允许从经过网络 12 的任何地方存取新的邮件。

30 在图 1 中，电子消息被分为两部分，即元数据部分和内容数据部分。元数据部分包括用户、消息、时间、日期、地址以及其他标识信息，内容数据部分包括与元数据部分相关的内容。内容数据和元数据由指针连接，这些指针与元数据一起存储，并指向其中存储有相关内容的内容数据存储器 33 中的位置。

在图 1 中的电子消息传递系统中，用户 11 连接到网络 12，网络 12 再

连接到传递器 13，传递器 13 再连接到存储服务器 14。认证/查询服务器 15 经过连接 19 连接到传递器 13。在一定情况下，网络 12 旁路传递器 13 并直接经过连接 10 连接到存储服务器 14，在此情况下，认证/查询服务器 15 也经过连接 18 连接到存储服务器 14。

5 在图 1 中，传递器 13 是一个接受来自用户的登录的透明代理，它使用认证/查询服务器 15 对用户进行身份认证，并且如果认证成功，向存储服务器 14 透明代理用户命令，并向用户代理来自存储服务器 14 的响应。存储服务器 14 存储位于服务器单元 26 内的用户邮箱，服务器单元 26 包括存储服务器单元 26-1、...、26-SU，每个服务器单元内部包括一个消息元数据存储器 32、一个消息内容数据存储器 33 以及一个服务器单元管理器 31。用户的
10 每个个人邮箱分布在元数据存储器 32 和消息内容数据存储器 33 之中，并在服务器单元管理器 31 的控制下被存取。用户的每个个人邮箱不分布在不同的服务器单元 26-1、...、26-SU 中。服务器单元 26 之间的用户的协调是在存储控制 16 的控制之下。

15 每个服务器单元 26 内的服务器单元管理器 31，管理消息元数据存储器 32 和消息内容数据存储器 33 内数据的存取，以及引起数据被检索或修改的服务协议命令。服务器单元管理器 31 协调对同一邮箱的多个请求。服务器单元管理器 31 是一个程序、线程或计算实体，功能是管理包括元数据存储器 32 的服务器单元 26 的地址空间。地址空间对协议服务器 34（包括每个
20 协议服务器 34-1、34-2 以及 34-3，见图 3）是公共的，对存储服务器单元 26 内存取的位置并不需要任何锁定协议。另外，每个存储服务器单元 26 是有效的，在于当开放用于存取物理地址的偏置表或其他机制时，因为地址空间是在服务器单元 31 的公共控制之下的，它们可以保持开放状态。

25 在图 1 的电子消息传递系统，使用简单邮件传送协议（SMTP）发送互联网消息，POP 和 IMAP 都使用 SMTP 发送消息。以相似的方式，存取和更新个人设置信息归于各自相伴的协议。

 在图 1 的电子消息传递系统中，当具体用户文件夹中的消息在贯穿那个文件夹的生命周期是唯一识别时，断线操作与在线操作有同样的要求，于是客户端和服务端可以周期性地重新同步具体消息的状态。

30 使用 POP、IMAP 和 SMTP 的电子消息传递系统-图 2

 图 2 更详细描述了图 1 的消息传递系统，其中，与互联网协议通信的消

息传递系统 4 使用 SMTP 协议, 用于经过连接 17D 的传递, 并使用 POP 和/或 IMAP 协议用于经过连接 17A 的存取。在图 2 中, 用户 11 包括虚拟用户 21 (例如经过互联网或其他远程网络连接的) 和本地用户 22 (例如经过局域网连接的), 于是网络 12 包括远程和本地网络 23。传递器 13 包括传递服务器 24 (使用 SMTP 协议) 和存取服务器 25 (使用互联网存取协议如 POP 或 IMAP)。存储服务器 14 包括存储服务器单元 26-1、...、26-SU 以及存储控制 16。认证/查询服务器 15 经过连接 19, 连接到存取服务器 25 和传递服务器 24。在网络 12 经过连接 10 而旁路传递器 13 的情况下, 服务器 15 也经过连接 18 连接到存储服务器 14。

图 2 的存储服务器 14 存储位于服务器单元 26 内的用户邮箱, 服务器单元 26 包括存储服务器单元 26-1、...、26-SU, 每个服务器单元 26 内部包括一个消息元数据存储器 32、一个消息内容数据存储器 33 以及一个服务器单元管理器 31, 如图 1 的存储服务器 14 的相关描述。服务器单元 26-1、...、26-SU 之间的协调是在存储控制 16 的控制之下的。在每个服务器单元 26 内, 服务器单元管理器 31 管理消息元数据存储器 32 和消息内容数据存储器 33 内数据的存取。

带有本地内容数据存储器的存储服务器单元-图 3

图 3 描述了图 1 和图 2 的消息传递系统内的典型存储服务器单元 26 的实施例。在图 3 中, 存储服务器单元 26 包括服务器单元管理器 31、元数据存储器 32 以及内容数据存储器 33。到和从服务器单元管理器 31 和内容数据存储器 33 的输入和输出来自协议服务器 34-1 (SMTP)、34-2 (POP) 以及 34-3 (IMAP)。协议服务器 34-1 (SMTP) 包括具有 SMTP 连接 17-1 的协议服务器单元 34-1₁、...、34-1_{u1}; 协议服务器 34-2 (POP) 包括具有 POP 连接 17-2 的协议服务器单元 34-2₁、...、34-2_{u2}; 以及协议服务器 34-3 (IMAP) 包括具有 IMAP 连接 17-3 的协议服务器单元 34-3₁、...、34-3_{u3}。协议服务器 34-1、34-2 以及 34-3 有到图 1 和图 2 的认证/查询服务器 15 的连接 18。

服务器单元 26 内的服务器单元管理器 31 管理消息元数据存储器 32 和消息内容数据存储器 33 内数据的存取, 和来自引起数据被检索或修改的协议服务器单元 34 的服务协议命令。存取用户邮箱的任何命令首先被负责存取元数据存储器 32 的存取的服务器单元管理器 31 处理。元数据存储器 32 中存储有到内容数据存储器 33 内的对应连接位置的指针, 消息的内容部分

在内容数据存储器 33 中存储或检索。服务器单元管理器 31 协调被不同的协议服务器单元 34 向同一邮箱的多个请求。用这种方式，服务器单元管理器 31 管理存储服务器单元 26 的地址空间。此地址空间对所有协议服务器 34-1、34-2 和 34-3 以及每个协议服务器单元 34-1₁、...、34-1_{U1}；协议服务器单元 5 34-2₁、...、34-2_{U2}；以及协议服务器单元 34-3₁、...、34-3_{U3} 是公共的。

带有远程内容数据存储器的存储服务器单元-图 4

在图 4 中，存储服务器单元 26 包括协议服务器 34-1、34-2 和 34-3、服务器单元管理器 31、元数据存储器 32 以及作为包括远程服务器 43 的远程数据存储器 33' 的一部分的内容数据存储器 33。远程数据存储器 33' 通过 10 网络 42，被连接到接口单元 34。服务器 43 和数据存储 45 之间的协议是例如 NFS 或任何其他的文件系统协议。

电子消息传递系统细节-图 5

在图 5 中，多个用户 11 被组织成包括用户组 11-1、...、11-U 的组。用户 11 连接到包括网络 12-1、12-2...、12-N 的网络 12。网络 12 再用 POP/IMAP 连接连接到存取服务器 13，以及用 SMTP 连接连接到输出传递服务器 51 和 15 输入传递服务器 52。存取服务器包括存取服务器单元 13-1、13-2、...、13-S。存取服务器 13 用 POP/IMAP 协议连接到存储服务器 14。存储服务器 14 包括存储服务器单元 14-1、14-2、...、14-SS 以及存储控制 16。输出传递服务器 51 包括输出传递服务器单元 51-1、51-2、...、51-OS，输入传递服 20 务器 52 包括输入传递服务器单元 52-1、52-1、...、52-IS。

在图 5 中，存储服务器单元 14-1、14-2、...、14-SS 提供到输出传递服务器 51 和输入传递服务器 52 的 SMTP 连接。在图 5 中，输出传递服务器 51 包括输出传递服务器单元 51-1、51-2、...、51-OS，输出传递服务器单元 51-1、51-2、...、51-OS 作为输入连接到网络 53 以及作为输入连接到传递服务器 52， 25 及分别连接到输入传递服务器单元 52-1、52-1、...、52-IS。

进入输出传递服务器 51 的通信是通过 SMTP 协议的，该通信包括来自网络 12、存储服务器 14 以及输入传递服务器 52 的通信。

在图 5 中，包括输入传递服务器单元 52-1、52-1、...、52-IS 的输入传递服务器 52 从输出传递服务器 51、网络 53、网络 12 以及存储服务器 14 接收 SMTP 协议输入，输入传递服务器 52 向存储服务器 14 以及输出传递服务器 30 51 传递 SMTP 协议通信。

在图 5 中, 也许包括如互联网的远程网络或本地网的网络 53, 连接到其他用户 54。

多用户组类型-图 6

图 6 描述图 1、图 2 以及图 5 的用户 11 的实现方式。图 6 中, 用户 11 以不同用户类型分组, 包括用户组 11-1 是属于电话类型。用户组 11-2 是传真类型。用户组 11-3 是组件类型。用户组 11-4 是电子邮件类型。在用户种类内可以包括任何其它组类型, 可被一般指定为用户组 11-T 用于指定其它用户类型。

服务器单元管理器公共单元-图 7

在图 7 中, 服务器单元管理器 31 包括一个公共存取控制单元 7-1 以及一个公共寻址单元 7-2。公共存取控制单元 7-1 接收来自协议服务器单元的消息, 单元 $34-i_j$ 是典型的协议服务器 34, 并对那些消息有公共的处理过程。在内容数据存储器 33 内将邮箱位置存取到公共寻址单元 7-2 是在控制下进行的, 其中位置 LOC_k 是典型的。因为内容数据存储器 33 内的所有邮箱的存取是在服务器单元管理器 31 的公共控制之下, 协议服务器 34 内的任何一个具体的消息服务器单元 $34-i_j$ 的错误并不会中断整个存储服务器单元 14。

服务器单元管理器细节-图 8

在图 8 中, 显示了服务器单元管理器 31 的更详细情况。服务器单元管理器 31 包括一个公共存取处理器 81, 处理从协议服务器 34 接收到的消息, 而与它们来自哪个协议服务器单元无关。特别地, 来自协议服务器 34 的一组消息被指定为 MSG_{A1} 、 MSG_{A2} 、...、 MSG_{Am} 、...、 MSG_{AM} 。这些消息的每一个被处理器 81 用包括例如锁定算法 82、开放算法 83 以及标志算法 84 的公共算法处理。公共存取处理器 81 通过执行公共算法, 对内容数据存储器 33 内的不同邮箱 87 判定控制状态, 其中邮箱 87-1、87-2、...87-M 分别被指定为 Mbox1、Mbox2...、MBoxM。特别地, 控制状态被存储在图 8 中的分别指定为 85-1、85-2、...、85-M 的 $MBox_1CTRL$ 、 $MBox_2CTRL$ 、...、 $MBox_MCTRL$ 中。此状态被公共寻址单元 7-2 用于控制内容数据存储器 33 内的邮箱 87 的存取。公共寻址单元 7-2 包括一个偏置控制 8-0, 其功能为: 作为控制状态 85-1、85-2、...、85-M 的功能, 对内容数据存储器 33 内的邮箱位置, 控制偏置地址的计算, 尤其那些与消息 MSG_{A1} 、 MSG_{A2} 、...、 MSG_{Am} 、...、 MSG_{AM} 相关使用的特别邮箱。

特别地, MSG_{A1} 被公共存取控制单元 7-1 处理, 来例如在 $MBox_1OFFSET$ 8-1 中创建偏置地址。相似的, MSG_{A2} 有一个偏置地址存储在例如指定为 8-M 的 $MBox_MOFFSET$ 中, 以及 MSG_{AG} 有一个例如在指定为 8-2 的 $MBox_2OFFSET$ 中创建的偏置。在公共存取控制单元 7-1 判定的合适存取条件下, 消息 MSG_{A1} 、 MSG_{A2} 以及 MSG_{AG} 分别存取内容数据存储器 33 内的邮箱 $MBox_1$ 、 $MBox_M$ 以及 $MBox_2$ 。服务器单元管理器 31 使用内容数据存储器 33 的文件系统操作, 使来自协议服务器 34 的消息与内容数据存储器 33 内的邮箱 87 相联系。

锁定。公共存取处理器 81 与文件系统一起, 执行锁定算法 82, 来判定内容数据存储器 33 内的被寻址邮箱是否可用于存取。假定典型的消息 MSG_{AG} 是存取邮箱 $MBox_2$, 那么锁定算法 82 得到一个锁定, 并将它存储在 $MBox_2CTRL$ 中, $MBox_2CTRL$ 在需要通过消息 MSG_{AG} 存取的时间段内, 锁定邮箱 $MBox_2$ 。

图 8 中, 如果在消息 MSG_{AG} 正在存取邮箱 $MBox_2$ 时, 另一个消息例如 MSG_{A1} 也试图存取邮箱 $MBox_2$, 那么锁定算法 82 阻止 MSG_{A1} 得到锁定, 并使得消息 MSG_{A1} 等待直到 MSG_{AG} 的锁定被消除。

总的来讲, 存储服务器 14 的功能如下:

- SMTP 将接收到的消息 MSG_m 传递到协议服务器单元;
- 协议服务器单元将接收到的消息传递到服务器单元管理器;
- 协议服务器单元等待得到对寻址邮箱的锁定;

当得到锁定时, 协议服务器对接收到的消息, 在 $UIDLIST$ (UID 列表) 上添加新的 UID;

- 接收到的消息被传递后, 协议服务器对被寻址邮箱解除锁定;
- 协议服务器单元发布返回到 SMTP 消息;
- 公共锁定算法寻找任何其它的终止, 如找到, 消除适合的锁定。

公共存取处理器 81 内的锁定算法 82, 功能是对所有的协议服务器寻找其它终止, 并根据检测结果, 自动消除相关邮箱的锁定。任何消息信道的协议服务器的终止, 都将引起锁定算法感应此中断, 并消除相应邮箱的锁定, 使得它对其它消息可用。以这种方式, 任何特别的协议服务器单元 $34-i_j$ 的错误, 将不会挂起那个单元最后连接的邮箱。在这种情况下, 锁定算法 82 的公共处理过程确保了所有系统的可靠性, 因为单个协议服务器单元不能挂起邮箱并使得它们长时间不可存取。

锁定的操作和消除锁定的操作由以下的顺序表示, 其中被中止 (abort)

的连接引起了一个处理碰撞 (crash)。

被中止的连接

POP: 从管理器获得事务 ID (TXN)

POP: 从 MAILBOXDB (邮箱 DB(数据库)) 的 UID 列表中消除 UID (邮箱 DB

5 因此被锁定)

POP: 处理碰撞

MGR: 检测丢失的连接

MGR: 中止 TXN, 解除数据库邮箱 DB 的锁定

10 开放。如果对被寻址邮箱到内容数据存储器 33 的存取连接没有开放, 那么开放算法 83 与文件系统一起作用, 来开放对寻址邮箱的连接。开放算法 83 将邮箱连接的开放状态保持需要消息存取的时间长短。另外, 对改进的操作, 开放算法 83 保持更长的时间。例如开放算法 83 基于最新使用的标准, 保持对邮箱连接的开放。最近使用的邮箱因此可被任何新的消息快速用于存取, 如果新的消息是对于任何开放的邮箱, 不需要使用文件系统重新创建连接。当需要对那些没有开放的邮箱创建新的连接时, 那么对最近最少使用的邮箱的连接被丢弃, 为新的连接制造空间。

20 开放算法 83 在邮箱消息队列 88 中, 存储开放邮箱的列表。每当新消息如典型新消息 MSG_{Am} 被公共存取处理器 81 处理时, 开放算法 83 将新邮箱 ID 推送到队列的顶端, 并参考队列 88 上的其它条目 (entry) 来判定对新消息 MSG_{Am} 的邮箱目前是否开放。如果开放, 在内容数据存储器 33 内存在一个对寻址邮箱的直接连接。如果没有开放, 那么开放算法 83 得到一个连接。对每个添加到队列顶端的新的邮箱, 其它在队列内的开放邮箱被向下推送。每当队列 88 达到极限时, 一个邮箱被从队列中清除, 为新的邮箱制造空间。典型地, 队列中最近最少使用的邮箱被清除, 为新的邮箱制造空间。对内容数据存储器 33 内邮箱的连接的关闭, 与来自协议服务器 34 的返回消息无关。清除算法的使用与返回消息无关, 使得系统比较协议服务器均基于消息的接收和返回开放和关闭连接的情况, 更加有效。

一个开放操作的举例出现在以下部分三 IMAP, 在子部分 “2. 选择” 下。

30 标志。对内容数据存储器的其它存取条件是在标志算法 84 的控制之下的, 标志算法 84 对被来自协议服务器 34 的消息用于对内容数据存储器控制并增强存取的各种不同条件设置标志。

一个标志操作的举例出现在以下部分“三 IMAP”，在子部分“6. 存储”下。

第一条消息-图 9

在图 9 中，如图 8 中的 MSG_{A1} 的一个典型的消息，被显示为消息号 40，
5 128 字节大小。图 9 的消息如下：

到: Sara Brown <sara@example2.org>
自: Bob Jones <bjones@example1.org>
主题: 取消
日期: 08/11/99
10 主体: 请取消会议...

图 9 的消息作为新的 UID 40 被存储，当前在邮箱 DB 数据库中有 UID 为 34 和 37 的消息。UID 40 的 HEADERDB (报头 DB) 数据库在“主题”字段存储“取消”，在“来自”字段存储“Bob Jones <bjones@example1.org>”，在“日期”字段存储“08/11/99”。MESSAGE DB (消息 DB) 在“大小”字段存储“128”，并且“标志”字段为空。LISTDB (列表 DB) 将“读”和“写”
15 字段标记为激活，表示此邮箱可用于读和写。消息的主体被存储在内容数据存储设备中例如/usr/sara/40 的位置。

第二条消息-图 10

在图 10 中，如图 8 中的 MSG_{A2} 的一个典型的消息，被显示为消息号 43，
20 256 字节大小。图 10 的消息如下：

到: Sara Brown <sara@example2.org>
自: MarySmith <msmith@example1.org>
主题: 新
日期: 08/12/99
25 主体: 你将找到新的文件在...
http: //www.example1.com ...

图 10 的消息作为新的 UID 43 被存储，当前在邮箱 DB 数据库中有 UID 为 34、37 和 40 的消息。对 UID 43 的报头 DB 数据库在“主题”字段存储“新”，在“来自”字段存储“MarySmith <msmith@example1.org>”，在“日期”字
30 段存储“08/12/99”。消息 DB 在“大小”字段存储“256”，并“标志”字段为空。列表 DB 将“读”和“写”字段标记为激活，表示此邮箱可用于读和

写。消息的主体被存储在内容数据存储器中例如/usr/sara/43的位置。

操作

POP 会话。POP 仅服务一个邮箱（输入邮箱）。在登录会话中，用户（例如，姓名=sbrown，密码=funfun）将映射到一个特定的邮箱（被系统内部管

5 理）。POP 也不允许多个用户同时存取同一邮箱。

登录

用户对传递器 13 创建连接。

用户向传递器 13 发送姓名、密码。

传递器 13 对比认证/查询服务器 15，对（姓名、密码）进行身份验证。

10 如果认证失败，传递器 13 响应于协议指定的错误，并丢弃连接（用户必须从开始重试登录）。

如果认证成功，传递器 13，具体是存取服务器 25，存取主导（host）用户特定邮箱的特别存储服务器单元 14-1、...、14-SS 中特定的一个。存取服务器 25 透明创建到特别存储服务器单元 14 的连接（或重新使用已存在连接）。

15 该连接在会话器件持续保持。

用户作一个 STAT 命令（来判定邮箱中消息的数目以及它们累积的大小）。

传递器 13 接受 STAT 命令，并转递给主机存储服务器单元 14。

主机存储服务器单元 14 作以下两个事情中的一个：

20 如果在元数据存储器 32 中，信息以预计算形式可用，主机存储服务器单元 14 检索它并将它送往传递器 13。

否则，主机存储服务器单元 14 基于每条信息，从元数据存储器 32 检索消息元数据信息。主机存储服务器单元 14 计算到 STAT 的响应，并响应传递器 13。

25 传递器 13 向用户转递响应。

用户向传递器 13 作出一个 LIST 命令（列出所有/特定消息 ID 以及它们各自的大小），传递器 13 将它转递给存储服务器单元 14。存储服务器单元 14 在元数据存储器 32 中，查找消息数和它们的大小，并将详细情况传给传递器 13，

30 传递器 13 再传给用户。

用户作出一个 RETR（检索被指定为命令的变量（argument）的消息）。

传递器 13 接受 RETR，并将它转递给主机存储服务器单元 14。

对变量中指定的每条消息，主机存储服务器单元：

首先检查发现此消息是否被标记为删除。如果是这样，向传递器 13 返回一个错误。

- 5 识别内容数据存储器 33 中检索消息内容数据的位置。从内容数据存储器 33 检索此消息的内容数据。这个检索过程可经过本地文件系统发生或经过网络（例如，使用 NFS 或 SQL）发生。然后将此检索数据向传递器 13 发送。

传递器 13 向用户转送此响应。

- 10 用户作出 DELE（删除指定为命令变量的特别消息）命令。

传递器 13 接受此命令，并将它和变量转递给主机存储服务器单元 14。

对变量中指定的消息，存储服务器单元检查元数据存储器 32，查看消息是否已被标记为删除。如果是，返回错误，如果不是，对在元数据存储器 32 内的消息标记删除字段。

- 15 用户作出 NOOP

传递器 13 直接响应于 NOOP（即，没有任何到/来自存储服务器 14 的附加通信量），或者向存储服务器单元 14 转递此命令。

存储服务器单元 14 不经过任何查找，响应 OK。

用户发布 RSET 命令

- 20 传递器 13 向主机存储服务器单元 14 单元转递此命令。

存储服务器单元 14 查找元数据存储器并识别标记为删除的消息。

如果找到，对每条标记为删除的消息消除删除字段的标记。然后响应 OK。

用户发布 QUIT

传递器 13 向主机存储服务器单元 14 转递此命令。

- 25 存储服务器单元 14 查找元数据存储器 32 中标记为删除的消息。然后从内容数据存储器 33 以及元数据存储器 32 中相应的条目消除每条这样的消息。

如果在消除删除消息中遇到问题，存储服务器单元 14 返回一个错误。否则返回 OK。

- 30 传递器 13 转递响应。

用户发布 UIDL

传递器 13 向主机存储服务器 14 转递 UIDL 命令和变量（若有的话）。

存储服务器单元 14 在元数据存储器 32 中，查找唯一的 UID（如果没有变量则对所有；或对特定变量）。然后用指定数据对请求作出响应。

传递器 13 转递数据。

- 5 IMAP 会话。存储服务器单元 14 在元数据存储器 32 中，含有用户文件（邮箱、存取控制列表以及邮箱许可、元数据的索引）。有一个每用户元数据存储器 13，含有关于用户已处理（读/删除/等）的消息的信息。

用户发布 LOGIN

- 10 传递器 13 认证。作为认证的一部分，传递器 13 判定主导用户特定邮箱的特别存储服务器单元，并创建到主机存储服务器单元 14 的连接。如果认证失败，连接被拆除。

用户发布 SELECT（选择邮箱）

传递器 13 向主机存储服务器单元 14 转递此命令。

- 15 存储服务器 14 开放来自元数据存储器 32 的邮箱信息。如果有问题，返回错误。否则返回 OK。

传递器 13 向用户转递响应。

用户发布 FETCH（选择报头信息；例如最近 50 个消息的报头等）

传递器 13 转递报头信息。

存储服务器单元 14 从元数据存储器 32 检索信息并响应。

- 20 传递器 13 向用户转递信息。

用户发布 FETCH（选择的消息主体，例如消息 51 的主体）

传递器 13 向主机存储服务器单元 14 转递报头信息。

- 25 存储服务器单元 14 从元数据存储器 32 中的一个消息 ID 字段，判定消息主体在内容数据存储器 33 中的位置，并从内容数据存储器 33 检索消息主体，并响应包括主体。

传递器 13 向用户转递消息主体。

用户发布 STORE FLAGS/DELETED（选择的消息 ID）

传递器 13 向主机存储服务器单元 14 转递删除（DELETE）。

- 30 存储服务器单元 14 在元数据存储器 32 中，标记消息的删除字段。响应错误或 OK。

传递器 13 向用户转递响应。

用户发布 EXPUNGE (发布清除信号来消除删除的消息)

传递器 13 向主机存储服务器单元 14 转递 EXPUNGE。

存储服务器单元 14 对 UID 列表中的每个 UID, 在消息 DB 中指定对此 UID 记录的删除, 并从邮箱 DB 的 UID 列表中指定消除此 UID, 并发送清除信号
5 启动这个邮箱的清除。

消息以 SMTP 本地 (native) 的形式存储。存储服务器单元不需要为了对用户请求作出响应, 从存储在内容数据存储器 33 中的格式, 对消息做任何计算或重新格式化。这个操作提供了高吞吐量。

同样, 元数据存储器 32 为特殊消息的请求/动作被最优化。例如为了标
10 记消息为“被删除”, 典型数据库需要几个命令, 其中, 在所述实施例中, 仅需要一个命令。

I. 被 POP/IMAP 使用的数据库

1. 列表 DB: 所有有效邮箱名的列表, 与每个邮箱相关的存取控制列表, 以及配额信息 (使用多少千字节, 允许的最大使用量)
- 15 2. 邮箱 DB: 与每个邮箱相关的数据库。它最主要包含: 这个邮箱中的 UID 列表。另外, 对每个存取此邮箱的用户, 存储的 UID 列表是用户已看过的 (已读到相关信息)。注意: UID 只是邮箱中与特定消息相关的唯一标识。
- 20 3. 消息 DB: 与每个邮箱相关的数据库。对于与邮箱中消息相关的每个 UID, 它存储那条消息的尺寸和一些标志 (例如删除标志) 以及每条消息的大小。
4. 报头 DB: 与每个邮箱相关的数据库。对于与邮箱中消息相关的每个 UID, 它存储“主题”字段, “来自”字段以及“日期”字段。

所有这些数据库通过所知“DB 服务器”的服务器存取, 确保了数据库
25 的完整性。

COPYRIGHT © 1999 Mirapoint, Inc

II. POP

1. LOGIN (using any of several authentication methods)

```

5  If user is successfully authenticated, then
    if mailbox has a "pop lock" then
        print error message
        return to authorization state
    end if

    place a "pop lock" on this mailbox
    get list of UIDS for each message in the mailbox from MAILBOXDB
10  MSGNO = 1

    For each uid in the list
        set SIZE = retrieve from MESSAGEDB the size of this message
        MESSAGES[MSGNO].uid = uid
        MESSAGES[MSGNO].size = SIZE
        MESSAGES[MSGNO].deleted = FALSE
        LASTMESSAGE = MSGNO
        MSGNO = MSGNO + 1
    end for
15  else
        print error message
        return to authorization state
    end if

```

2. STAT

```

20  set NMESSAGES = 0
    set NBYTES = 0
    I = 1

    while (I < MSGNO)
        if not MESSAGES[I].deleted then
            NMESSAGES = NMESSAGES + 1
            NBYTES = NBYTES + MESSAGES[I].size
        end if
        I = I + 1
    end while
25  print value of NMESSAGES and NBYTES

```

3. LIST

```

    if argument given to LIST
        MSG = parse argument
        if (MSG < 1 or MSG = MSGNO) then
            print error
30

```

COPYRIGHT © 1999 Mirapoint, Inc

```

5      else
          print value of MSG, MESSAGES[MSG].size
        end if
      else
        set I = 1
        while (I < MSGNO)
          if not MESSAGES[I].deleted then
            print value of I, MESSAGES[I].size
          end if
          I = I + 1
        end while
      end if
10
4. RETR

    MSG = parse argument to RETR
    if (MSG < 1 or MSG = MSGNO) then
      print error
    else
      if MESSAGES[MSG].deleted then
        print error
      else
15        print contents of file associated with MSG
      end if
    end if

5. DELE

    MSG = parse argument to RETR
    if (MSG < 1 or MSG = MSGNO) then
      print error
    else
20      set MESSAGES[MSG].deleted = TRUE
    end if

6. NOOP

    print OK

7. RSET
25

    I = 1
    while (I < MSGNO)
      set MESSAGES[I].deleted = FALSE
      I = I + 1
    end while

8. UIDL
30

```

COPYRIGHT © 1999 Mirapoint, Inc

```

5      if argument given to UIDL
        MSG = parse argument
        if (MSG < 1 or MSG = MSGNO) then
            print error
        else
            print value of MSG, MESSAGES[MSG].uid
        end if
    else
        I = 1
        while (I < MSGNO)
            if not MESSAGES[I].deleted then
10                print value of I, MESSAGES[I].uid
            fi
            I = I + 1
        end while
    end if

    9. QUIT (Note that any modification to a database, MAILBOXDB or MESSAGEDB,
    automatically causes a lock of the database and the lock remains until a COMMIT unlocks the
    database)

15      if no messages have been deleted with DELETE command
        print OK
        disconnect
    else
        set TXN = create transaction from DB server
        I = 1
        while (I < MSGNO)
            if MESSAGES[I].deleted then
20                using TXN, send to DB server the following work
                items:
                remove MESSAGES[I].uid from MAILBOXDB UID
                list
                remove from MESSAGEDB record associated with uid

                fi
                I = I + 1
            end while
            send COMMIT to DB server for this TXN
            if COMMIT fails
25                print error
            else
                send Clean signal to initiate cleaning of this mailbox
            end if
        end if
        remove "pop lock" from mailbox
    
```

30

III. IMAP

1. LOGIN (using any of several authentication methods)

```
5      if (authentication is successful) then
          set USER = authenticated user name
      else
          print error message
          return to authorization state
      fi
```

2. SELECT

```
10      MAILBOXNAME = parse argument to select
      retrieve access-control list (ACL) from LISTDB for MAILBOXNAME.
      if (mailbox doesn't exist or ACL doesn't permit USER to select) then
          print error
      else
          acquire reference to MAILBOXDB from manager, manager opens
              MAILBOX-NAME, if not already open, and places a reference
              on the open queue
          retrieve UIDLIST for MAILBOXNAME from MAILBOXDB.
          print number of messages in mailbox (size of UIDLIST)
              -- other information is looked up and printed (recent,
              unseen, invalidity, etc.)--
      end if
```

3. FETCH flags

```
20      FETCHUIDS = parse UIDS to fetch from mailbox
      for each uid in FETCHUIDS
          retrieve from MESSAGEDB flags for this UID in selected mailbox
          print flags
      end for
```

4. FETCH headers

```
25      FETCHUIDS = parse UIDS to fetch from mailbox
      for each uid in FETCHUIDS
          retrieve from HEADERDB headers for this UID in selected mailbox
          print headers
      end for
```

5. FETCH body

```
30      FETCHUIDS = parse UIDS to fetch from mailbox
      for each uid in FETCHUIDS
          open file associated with this UID in selected mailbox
          print contents of file
```

COPYRIGHT © 1999 Mirapoint, Inc

close file
end for

5

6. STORE flags (deleted)

STOREUIDS = parse UIDS to fetch from mailbox
TXN = create transaction from DB server
for each uid in STOREUIDS
 using TXN, set "deleted" flag for this UID in MESSAGEDB
end for
commit TXN
if (commit failed) then
 print error
else
 print OK
end if

10

7. EXPUNGE

if mailbox has a "pop lock" then
 print error message
 return to authorization state
end if
place a "pop lock" on this mailbox
create TXN
for each uid in UIDLIST
 using TXN, indicate delete of record for this UID in MESSAGEDB
 using TXN, indicate removal of this uid from UIDLIST in
 MAILBOXDB
end for
commit TXN
if (commit failed) then
 print error
else
 send Clean signal to initiate cleaning of this mailbox
 print OK
end if
remove "pop lock" from this mailbox

15

20

8. CLEANER

while true,
 do
 wait for Clean signal to clean MBOX.
 retrieve UIDLIST from MAILBOXDB for MBOX.
 For each file in MBOX's directory
 do
 if file not in UIDLIST
 delete file
 endif
 done
 done

25

30

本发明被特别示出并参考优选实施例描述，其中，本领域的技术人员将理解，在不脱离本发明的范围的情况下，可作出各种形式上以及细节的改变。

5

10

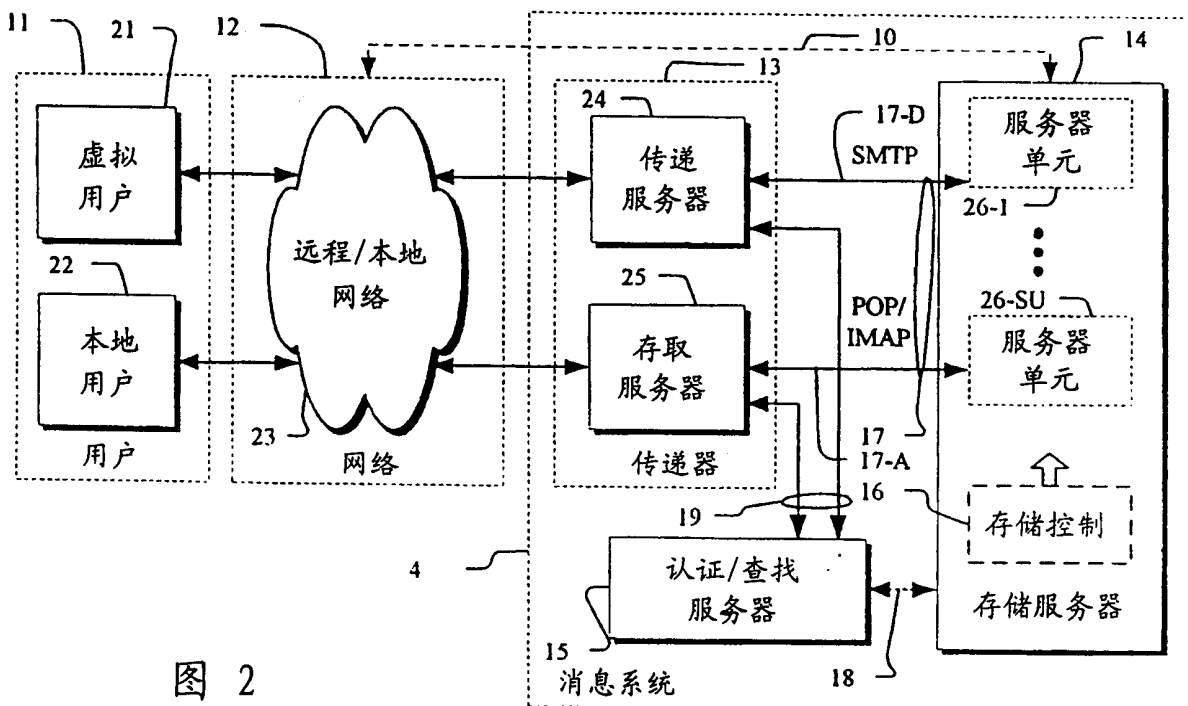
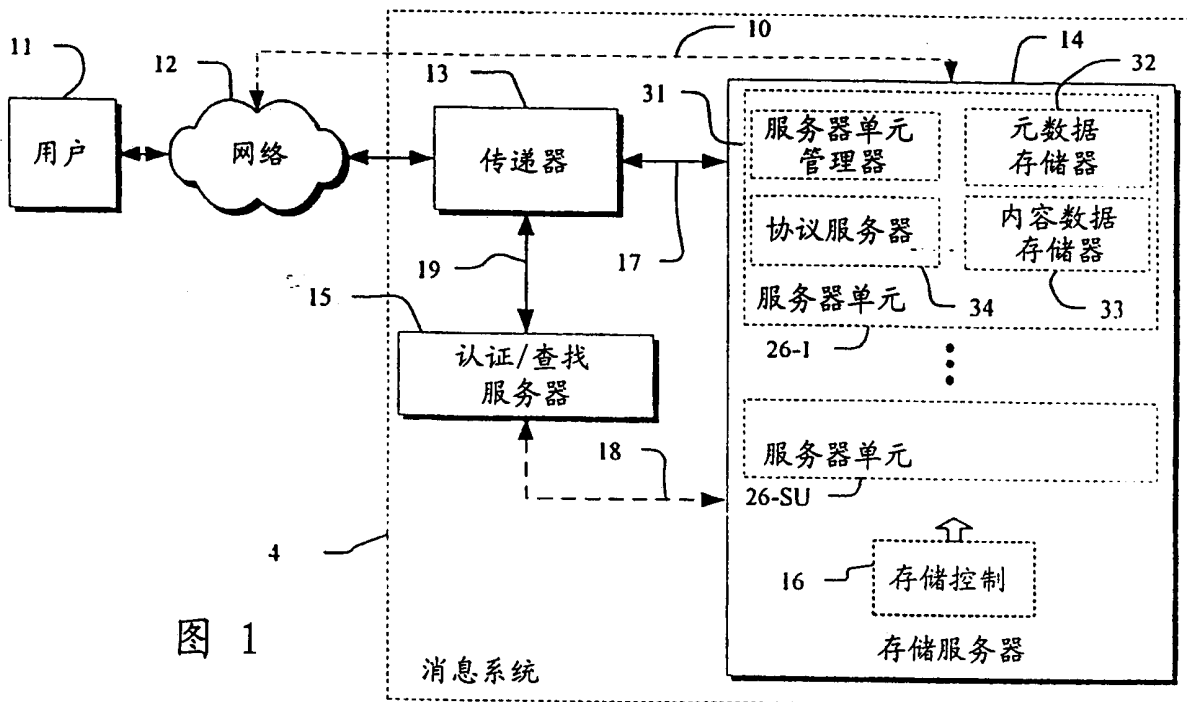
15

20

25

30

说明书附图



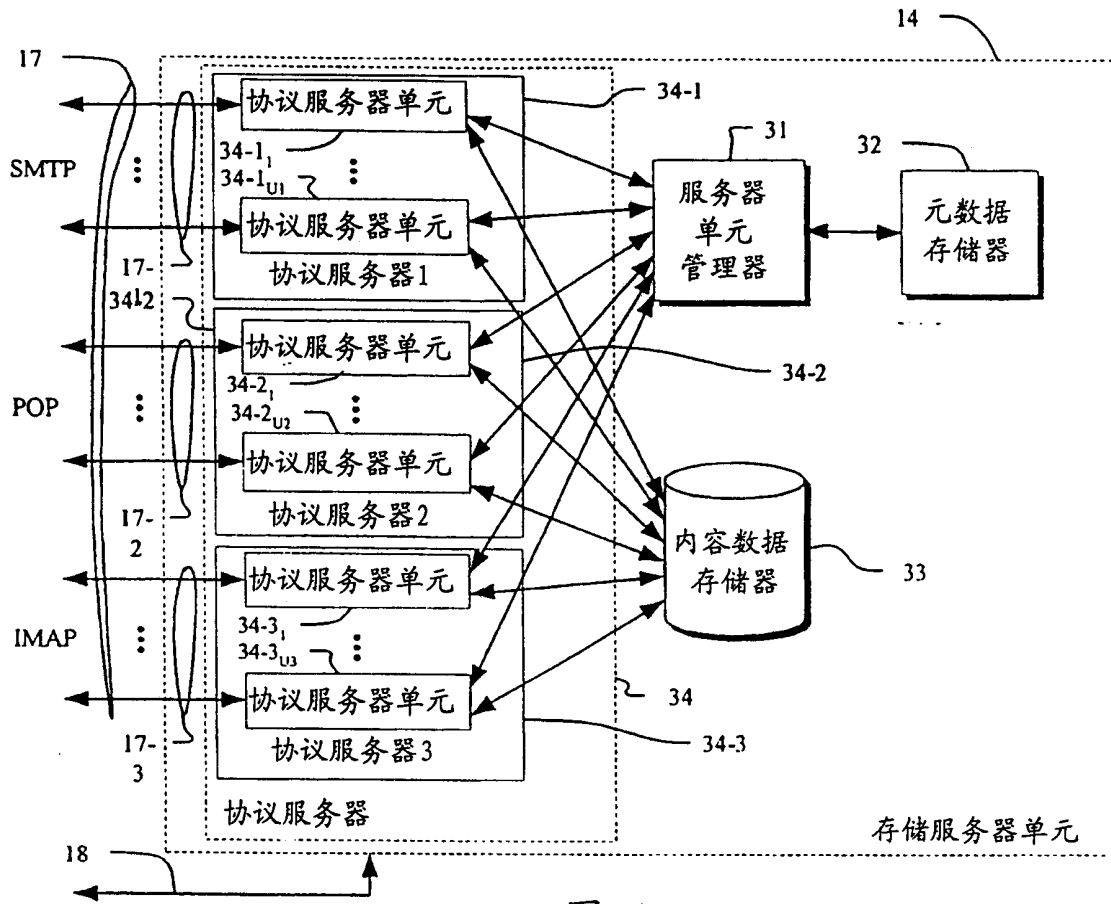


图 3

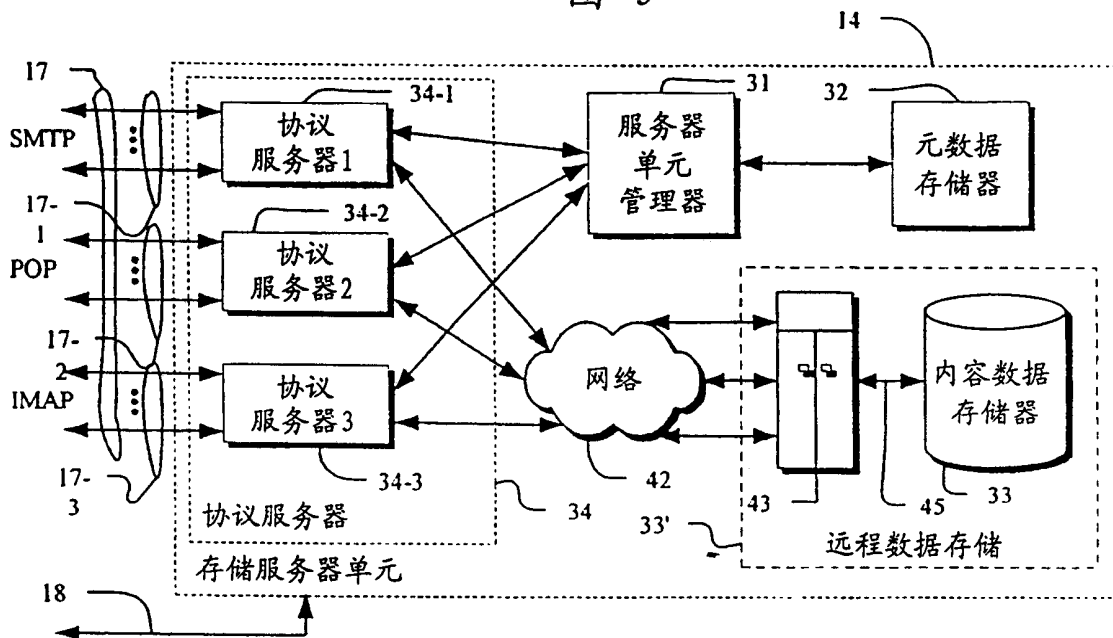


图 4

34

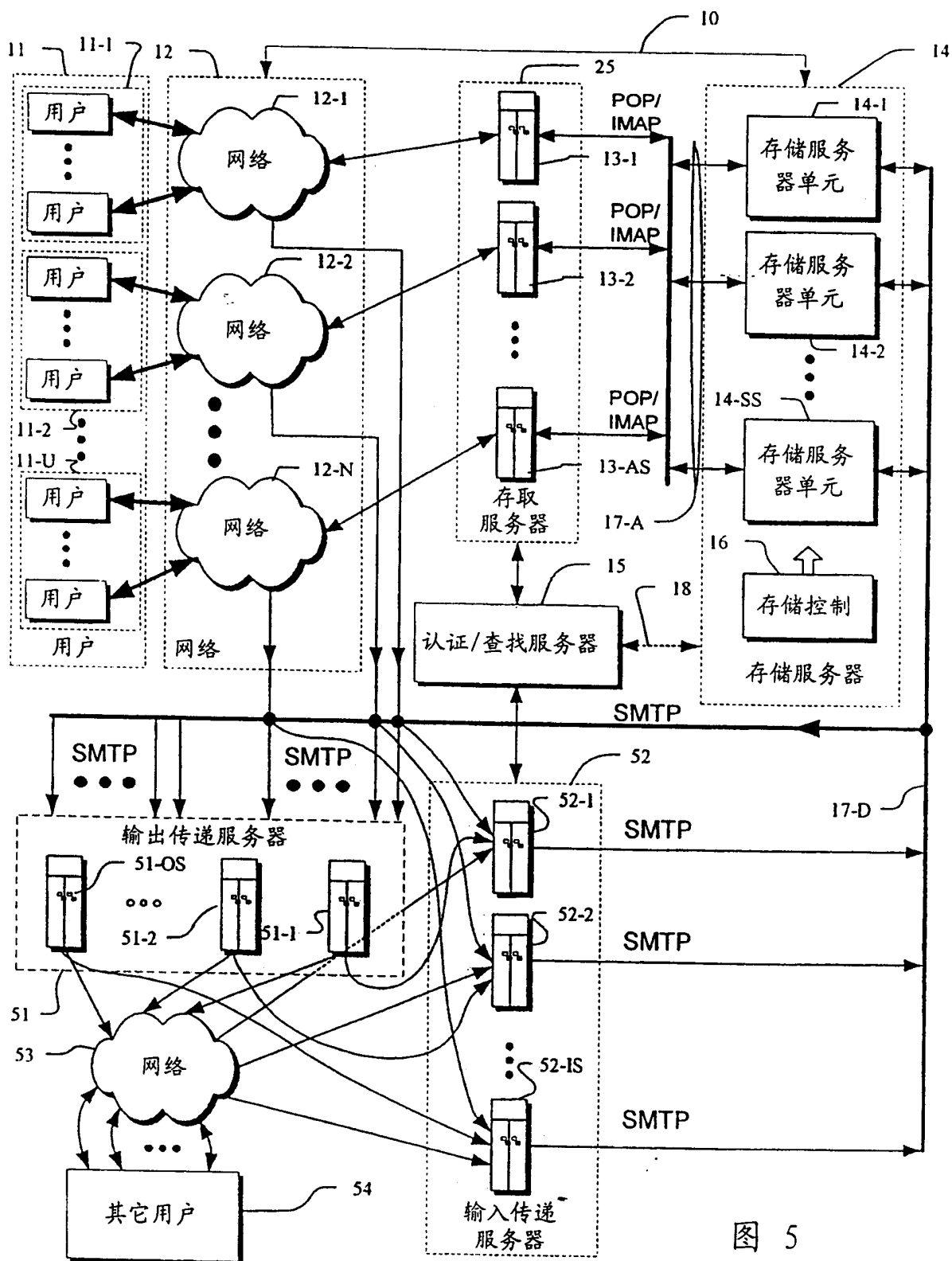


图 5

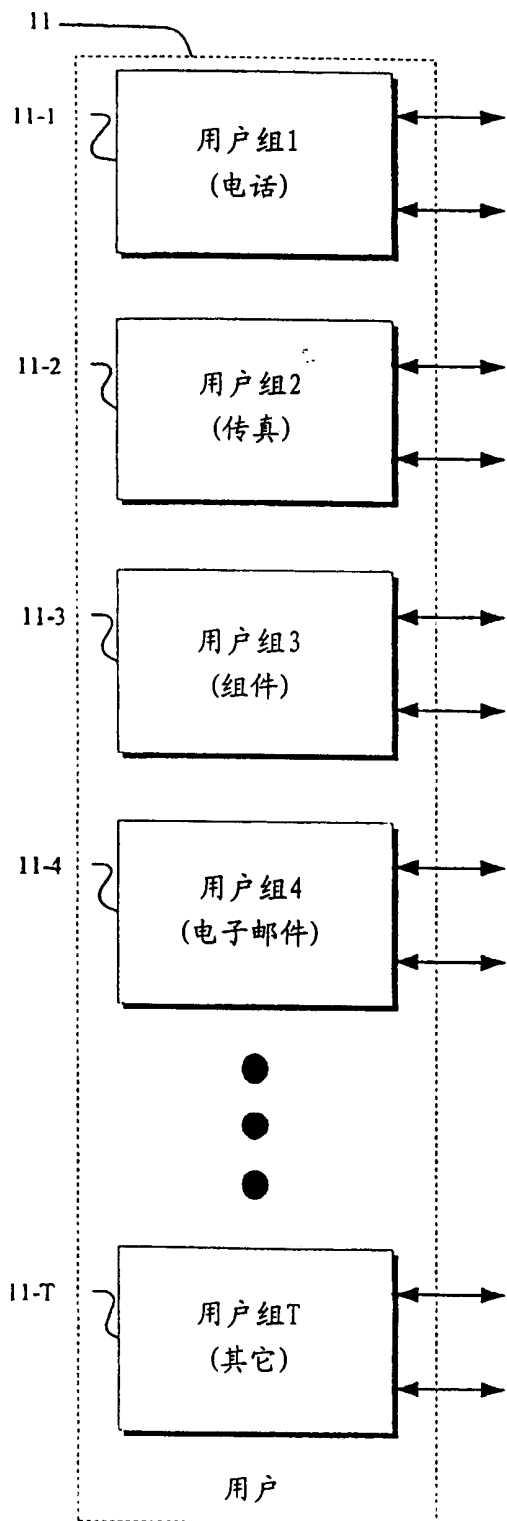


图 6

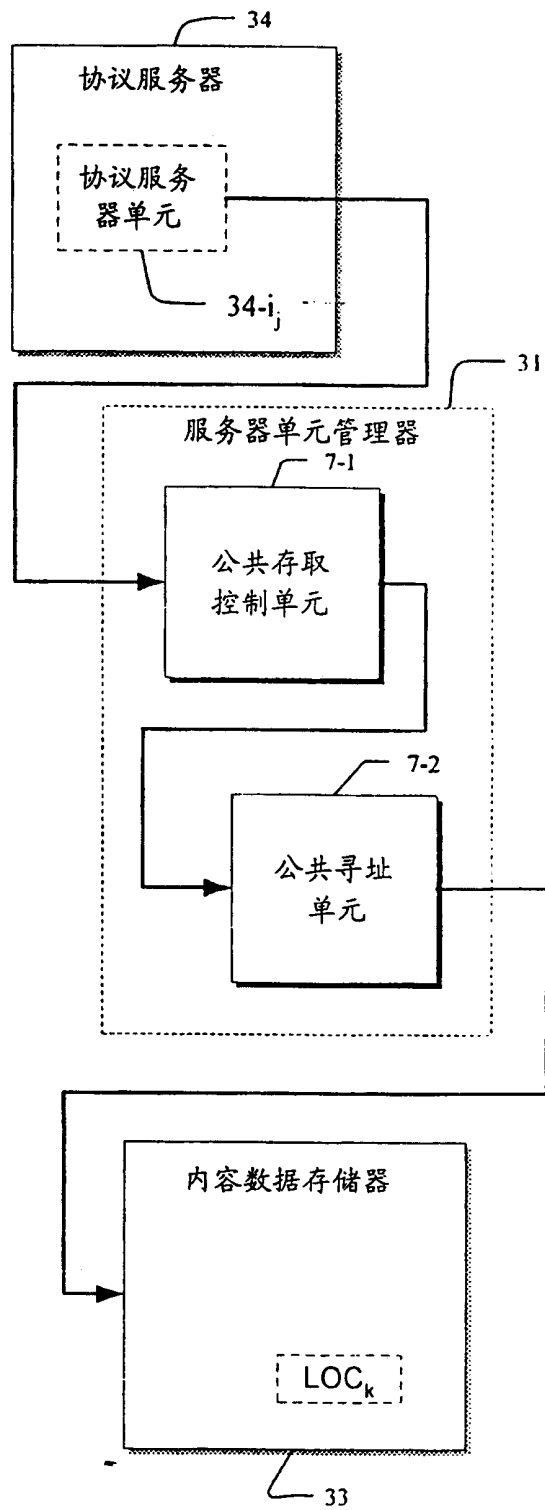


图 7

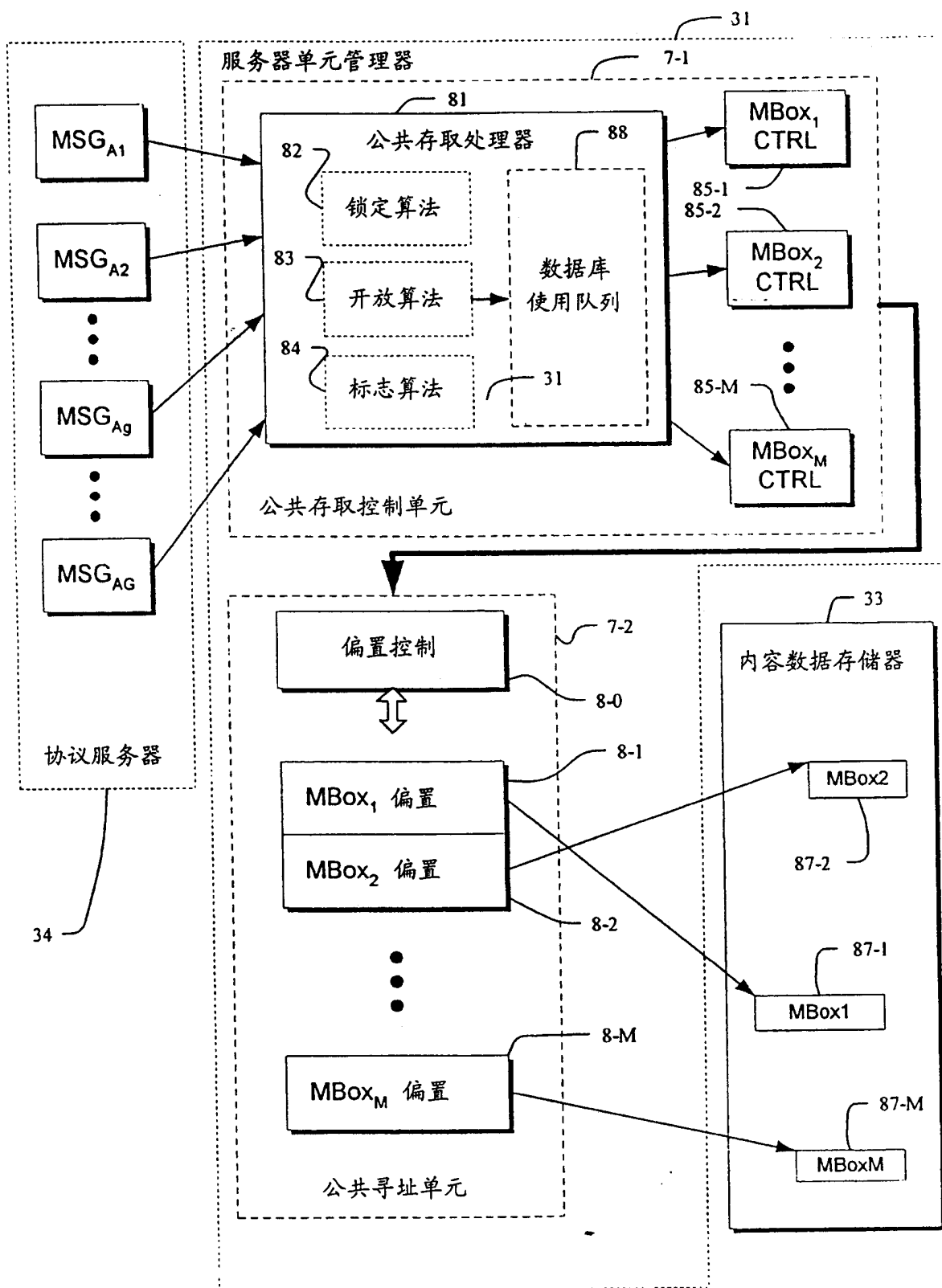


图 8

<u>邮箱DB</u>	<u>UID列表</u> 34 37	<u>新UID</u> 40			元
<u>报头DB</u> 34 37 40	<u>主题</u> 取消	<u>来自</u> Bob Jones<bjones@example1.org>	<u>日期</u> 08/11/99		
<u>消息DB</u> 34 37 40	<u>大小</u> 128	<u>标志</u>			
<u>列表DB</u>	<u>读</u> X	<u>写</u> X			
主体: 请取消会议。 . . .					内容

图 9

<u>邮箱DB</u>	<u>UID列表</u> 34 37 40	<u>新UID</u> 43			元
<u>报头DB</u> 34 37 40 43	<u>主题</u> 取消 新	<u>来自</u> Bob Jones<bjones@example1.org> Mary Smith <msmith@example2.org>	<u>日期</u> 08/11/99 08/12/99		
<u>消息DB</u> 34 37 40 43	<u>大小</u> 128 256	<u>标志</u>			
<u>列表DB</u>	<u>读</u> X	<u>写</u> X			
主体: 你将在 http://www.example1.com 找到新文件.					内容

图 10