



(12) 发明专利申请

(10) 申请公布号 CN 102687487 A

(43) 申请公布日 2012.09.19

(21) 申请号 201080059955.4

(74) 专利代理机构 北京市中咨律师事务所

(22) 申请日 2010.10.25

11247

(30) 优先权数据

12/651,089 2009.12.31 US

代理人 杨晓光 于静

(85) PCT申请进入国家阶段日

(51) Int. Cl.

2012.06.29

H04L 29/08(2006.01)

H04L 29/06(2006.01)

(86) PCT申请的申请数据

PCT/FI2010/050842 2010.10.25

(87) PCT申请的公布数据

W02011/080381 EN 2011.07.07

(71) 申请人 诺基亚公司

地址 芬兰埃斯波

(72) 发明人 付岩

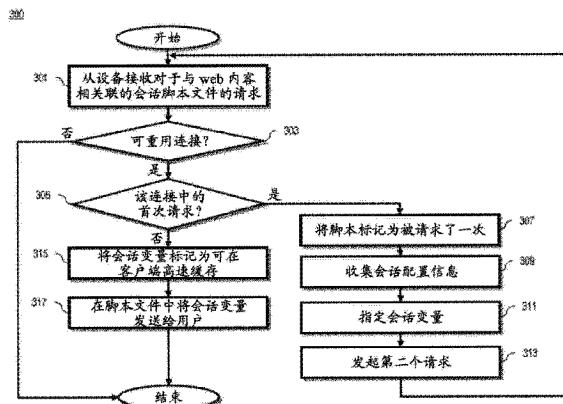
权利要求书 3 页 说明书 18 页 附图 10 页

(54) 发明名称

用于提供客户端高速缓存的方法和设备

(57) 摘要

提供一种用于在网络通信中提供客户端高速缓存的方法。客户端高速缓存模块从设备接收对关联于 web 内容的脚本文件的请求，所述脚本文件支持在所述设备处的客户端高速缓存。客户端高速缓存模块识别所述请求是经由支持连接重用的连接接收的。客户端高速缓存模块确定所述请求是否是首次请求。如果请求是首次请求，则客户端一致性模块从所述设备收集用于所述 web 内容的会话配置信息，基于所述会话配置信息在所述脚本文件中指定会话变量，并且经由所述连接发起对于所述脚本文件的后续请求。如果请求是后续请求，则客户端高速缓存模块将所述脚本文件中的会话变量标记为可在所述设备处高速缓存，并且至少部分地使得脚本文件向所述设备发送。



1. 一种方法,包括 :

从设备接收对于与 web 内容相关联的脚本文件的请求,所述脚本文件支持在所述设备处进行客户端高速缓存;

识别所述请求是经由支持连接重用的连接接收的;

确定所述请求是否是首次请求;

如果所述请求是首次请求,则从所述设备收集用于所述 web 内容的会话配置信息,基于所述会话配置信息在所述脚本文件中指定会话变量,以及,经由所述连接发起对所述脚本文件的后续请求;以及

如果所述请求是后续请求,则在所述脚本文件中将所述会话变量标记为可在所述设备处进行高速缓存,并且,至少部分地使得所述脚本文件向所述设备发送。

2. 根据权利要求 1 的方法,进一步包括 :

从所述设备接收对于与所述脚本文件相关联的 web 内容的另一请求;以及

至少部分地使得基于所述高速缓存的会话变量提供所述 web 内容。

3. 根据权利要求 1-2 中任一个的方法,其中,所述会话变量是所述设备本地的,并且仅适用于所述相关联的 web 内容。

4. 根据权利要求 1-2 中任一个的方法,其中,所述会话配置信息适用于多个域,所述方法进一步包括 :

在所述会话变量中指定所述多个域。

5. 根据权利要求 1-4 中任一个的方法,其中,在预定时间段之后,或在所述设备的用户请求时,所述脚本文件被从所述设备移除。

6. 根据权利要求 1-5 中任一个的方法,其中,所述连接是超文本传输协议(HTTP) Keep-Alive 连接,以及其中,对所述连接的所述识别是基于对应于所述 HTTP Keep-Alive 连接的传输控制协议(TCP)套接字标识符、工作进程标识符、线程标识符或其组合。

7. 根据权利要求 1-6 中任一个的方法,其中,所述脚本文件是 JavaScript 文件。

8. 一种设备,包括 :

至少一个处理器;以及

至少一个包括计算机程序代码的存储器,

所述至少一个存储器和所述计算机程序代码被配置为,通过使用所述至少一个处理器,使得所述设备实施至少以下,

从设备接收对于与 web 内容相关联的脚本文件的请求,所述脚本文件支持在所述设备处进行客户端高速缓存;

识别所述请求是经由支持连接重用的连接接收的;

确定所述请求是否是首次请求;

如果所述请求是首次请求,则从所述设备收集用于所述 web 内容的会话配置信息,基于所述会话配置信息在所述脚本文件中指定会话变量,以及,经由所述连接发起对于所述脚本文件的后续请求;以及

如果所述请求是后续请求,则在所述脚本文件中将所述会话变量标记为可在所述设备处进行高速缓存,并且至少部分地使得所述脚本文件向所述设备发送。

9. 根据权利要求 8 的设备,其中,进一步使得所述设备 :

从所述设备接收对于与所述脚本文件相关联的所述 web 内容的另一请求 ; 以及至少部分地使得基于所述高速缓存的会话变量提供所述 web 内容。

10. 根据权利要求 8-9 中任一个的设备, 其中, 所述会话变量是所述设备本地的, 并且仅适用于所述相关联的 web 内容。

11. 根据权利要求 8-9 中任一个的设备, 其中, 所述会话配置信息应用于多个域, 以及进一步使得所述设备 :

在所述会话变量中指定所述多个域。

12. 根据权利要求 8-11 中任一个的设备, 其中, 在预定时间段之后, 或在所述设备的用户请求时, 所述脚本文件被从所述设备移除。

13. 根据权利要求 8-12 中任一个的设备, 其中, 所述连接是超文本传输协议 (HTTP) Keep-Alive 连接, 以及其中, 对所述连接的所述识别是基于对应于所述 HTTP Keep-Alive 连接的传输控制协议 (TCP) 套接字标识符、工作进程标识符、线程标识符或其组合。

14. 根据权利要求 8-13 中任一个的设备, 其中, 所述脚本文件是 JavaScript 文件。

15. 根据权利要求 8-14 中任一个的设备, 其中, 所述设备是移动电话, 所述移动电话进一步包括 :

用户界面电路和用户界面软件, 被配置为, 促进用户通过使用显示器对所述移动电话的至少某些功能的控制, 以及被配置为, 对用户输入做出响应 ; 以及

显示器和显示电路, 其被配置为, 显示所述移动电话的用户界面的至少一部分, 所述显示器和显示电路被配置为, 促进用户对所述移动电话的至少某些功能的控制。

16. 一种计算机可读存储介质, 携带一个或更多一个或更多指令的序列, 所述指令当被一个或更多处理器执行时, 使得所述设备至少实施以下步骤 :

从设备接收对于与 web 内容相关联的脚本文件的请求, 所述脚本文件支持在所述设备处进行客户端高速缓存 ;

识别所述请求是经由支持连接重用的连接接收的 ;

确定所述请求是否是首次请求 ;

如果所述请求是首次请求, 则从所述设备收集用于所述 web 内容的会话配置信息, 基于所述会话配置信息在所述脚本文件中指定会话变量, 以及, 经由所述连接发起对所述脚本文件的后续请求 ; 以及

如果所述请求是后续请求, 则在所述脚本文件中将所述会话变量标记为可在所述设备处进行高速缓存, 并且, 至少部分地使得所述脚本文件向所述设备发送。

17. 根据权利要求 16 的计算机可读存储介质, 其中, 使得所述设备进一步实施 :

从所述设备接收对于与所述脚本文件相关联的所述 web 内容的另一请求 ; 以及

至少部分地使得基于所述高速缓存的会话变量提供所述 web 内容。

18. 根据权利要求 16-17 中任一个所述的计算机可读存储介质, 其中, 所述会话变量是所述设备本地的, 并且仅适用于所述相关联的 web 内容。

19. 根据权利要求 18 的计算机可读存储介质, 其中, 所述会话配置信息适用于多个域, 以及, 使得所述设备进一步实施 :

在所述会话变量中指定所述多个域。

20. 根据权利要求 16 的计算机可读存储介质, 其中, 所述连接是超文本传输协议

(HTTP) Keep-Alive 连接, 以及其中, 对所述连接的所述识别是基于对应于所述 HTTP Keep-Alive 连接的传输控制协议 (TCP) 套接字标识符、工作进程标识符、线程标识符或其组合。

21. 一种设备, 包括用于实施根据权利要求 1-7 中任一个的方法的装置。
22. 一种计算机可读存储介质, 携带一个或更多一个或更多指令的序列, 所述指令当被一个或更多处理器执行时, 使得设备至少实施根据权利要求 1-7 中任一个的方法。
23. 一种计算机程序产品, 携带一个或更多一个或更多指令的序列, 所述指令当被一个或更多处理器执行时, 使得设备至少实施根据权利要求 1-7 中任一个的方法。

用于提供客户端高速缓存的方法和设备

背景技术

[0001] 网络服务提供商和设备制造商继续受到通过例如提供拼写网络服务向消费者递送价值和便利的挑战。一个极大改进用户体验的网络特征是例如会话变量的客户端高速缓存,其中,所述会话变量为网络服务提供例如个性化、自动用户认证和其它形式会话持久化的特征。特别地,客户端持久化或高速缓存使客户端设备能够本地存储信息,其中,所述信息用于认证、会话跟踪(例如状态维护)等,以及用于记住关于用户的特定信息,例如服务习惯选择、浏览历史、之前的活动等。当用户开始服务会话(例如 web 浏览会话)并且然后在稍后的时间继续该会话时,该信息可以然后被用于提供用户服务体验上的连续性。然而,因为用于提供所述持久化的信息(例如登录证书、网站访问历史等)可能是敏感的,所以服务提供商和设备制造商面临重大技术挑战,所述挑战在于,在提供来自客户端高速缓存机制的增加的功能的同时,保护信息不被公开或恶意利用。

发明内容

[0002] 因此,存在对可替换方法的需求,所述可替换方法用于提供克服上面提到的限制的客户端高速缓存。

[0003] 根据一个实施例,一种方法包括,从设备接收对于与 web 内容相关联的脚本文件的请求,所述脚本文件支持在所述设备处进行客户端高速缓存。所述方法还包括,识别所述请求是经由支持连接重用的连接接收的。所述方法进一步包括,确定所述请求是否是首次请求。如果所述请求是首次请求,则所述方法包括,从所述设备收集用于所述 web 内容的会话配置信息,基于所述会话配置信息在所述脚本文件中指定会话变量,以及,经由所述连接发起对于所述脚本文件的后续请求。如果所述请求是后续请求,则所述方法还包括,在所述脚本文件中将所述会话变量标记为可在所述设备处高速缓存,以及至少部分地使得所述脚本文件向所述设备发送。

[0004] 根据另一实施例,一种设备,包括至少一个处理器和至少一个包括计算机程序代码的存储器,所述至少一个存储器和所述计算机程序代码被配置为,通过使用所述至少一个处理器,至少部分地使得所述设备从设备接收对于与 web 内容相关联的脚本文件的请求,所述脚本文件支持在所述设备处进行客户端高速缓存。还导致所述设备,识别所述请求是经由支持连接重用的连接接收的。进一步导致所述设备,确定所述请求是否是首次请求。如果所述请求是首次请求,则进一步导致所述设备,从所述设备收集用于所述 web 内容的会话配置信息,基于所述会话配置信息在所述脚本文件中指定会话变量,以及,经由所述连接发起对于所述脚本文件的后续请求。如果所述请求是后继请求,则进一步导致所述设备,在所述脚本文件中将所述会话变量标记为可在所述设备处高速缓存,以及至少部分地使得所述脚本文件向所述设备发送。

[0005] 根据另一实施例,一种计算机可读存储介质,携带一个或更多一个或更多指令的序列,所述一个或更多一个或更多指令的序列当被一个或更多处理器执行时,至少部分地使得设备从设备接收对于与 web 内容相关联的脚本文件的请求,所述脚本文件支持在所述

设备处进行客户端高速缓存。还导致所述设备，识别所述请求是经由支持连接重用的连接接收的。进一步导致所述设备，确定所述请求是否是首次请求。如果所述请求是首次请求，则进一步导致所述设备，从所述设备收集用于所述 web 内容的会话配置信息，基于所述会话配置信息在所述脚本文件中指定会话变量，以及，经由所述连接发起对于所述脚本文件的后续请求。如果所述请求是后继请求，则进一步导致所述设备，在所述脚本文件中将所述会话变量标记为可在所述设备处高速缓存，以及至少部分地使得所述脚本文件向所述设备发送。

[0006] 根据另一实施例，一种设备包括这样的装置，所述装置用于从设备接收对于与关联于 web 内容的脚本文件的请求，所述脚本文件支持在所述设备处进行客户端高速缓存。所述设备还包括这样的装置，所述装置用于识别所述请求是经由支持连接重用的连接接收的。所述设备进一步包括这样的装置，所述装置用于确定所述请求是否是首次请求。如果所述请求是首次请求，则所述设备进一步包括这样的装置，所述装置用于从所述设备收集用于所述 web 内容的会话配置信息，基于所述会话配置信息在所述脚本文件中指定会话变量，以及，经由所述连接发起对于所述脚本文件的后续请求。如果所述请求是后续请求，则所述设备包括这样的装置，所述装置用于在所述脚本文件中将所述会话变量标记为可在所述设备处高速缓存，以及，至少部分地使得所述脚本文件向所述设备发送。

[0007] 从以下详细描述中，简单地通过示出包括所设想的用于实现本发明的最佳模式的许多特定实施例和实现，本发明的仍然其它方面、特征和优势是显而易见的。全部在不脱离本发明的精神和范围的情况下，本发明还能够做出其它的并且不同的实施例，并且可以在各个显而易见的方面修改其若干细节。相应地，应当把附图和说明书看作本质上是示例性的，并且不是限制性的。

附图说明

- [0008] 在附图的图中，作为示例并且不作为限制示出了本发明的实施例，其中：
- [0009] 图 1 是根据一个实施例的能够提供客户端高速缓存的系统的图；
- [0010] 图 2 是根据一个实施例的客户端高速缓存模块的部件的图；
- [0011] 图 3 是根据一个实施例的向设备提供对于 web 内容的首个请求的客户端高速缓存的过程的流程图；
- [0012] 图 4 是根据一个实施例的用户从服务器请求 web 内容的过程的流程图；
- [0013] 图 5 是根据一个实施例的向设备提供对于 web 内容的后续请求的客户端高速缓存的过程的流程图；
- [0014] 图 6 是时序图，其示出了根据一个实施例的用于提供客户端高速缓存的消息序列和过程；
- [0015] 图 7 是根据一个实施例的示例性用户界面的图，所述用户界面用于提供与客户端高速缓存相关的数据；
- [0016] 图 8 是可以用于实现本发明的一个实施例的硬件的图；
- [0017] 图 9 是可以用于实现本发明的一个实施例的芯片组的图；以及
- [0018] 图 10 是可以用于实现本发明的一个实施例的移动终端(例如手机)的图。

具体实施方式

[0019] 公开了用于提供客户端高速缓存的方法、设备和计算机程序的示例。在以下描述中,为了说明目的,阐述了大量特定细节,以便提供对本发明的实施例的深入理解。然而,对于本领域的技术人员显而易见,可以在没有这些特定细节或具有等价安排的情况下实践本发明的实施例。在其它情况下,以框图形式示出了众所周知的结构和设备,以便避免不必要的使本发明的实施例晦涩难懂。

[0020] 图 1 是根据一个实施例的能够提供客户端高速缓存的系统的图。如前面指出的,许多应用在用户设备(即客户端)上而非仅在服务器上存储用于完成或继续应用(例如 web 应用)功能和任务的数据。这样,应用可以在需要时降低服务器负载并且在客户端上完成互联网任务,由此还减少网络带宽和资源使用。在用户设备处所存储或高速缓存的数据一般称为客户端持久数据。在一个实施例中,客户端持久数据是偶发连接计算(OCC)需求的解决方案,其中,所述偶发连接计算对于这样的体系结构或框架进行计算,所述体系结构或框架允许在未连接到互联网时运行 web 应用的某些方面。OCC 是基于以下观点的软件体系结构:终端用户应该能够甚至在临时被断开时或在无线连接出故障或否则不可用时继续与互联网应用一起运转。因此,对于将可在缺少互联网连接时运转的应用,该用于所需的数据需要被存储在客户现场设备上。

[0021] 另外,应当设想,当不存在网络连接,但服务或应用仍然优选或被配置为在客户设备上存储用于维持功能连续性的数据时,也可以使用 OCC 原理。历史上,使用 cookie 是最受欢迎的存储客户端持久化数据的方法中的一个。cookie(例如浏览器 cookie 或超文本传输协议(HTTP) cookie)一般是由例如 web 浏览器或其它应用存储在用户设备上的少量文本。cookie 由一个或更多包含有限位数信息的名值对组成,所述信息例如是用户习惯选择、购物车内容、基于服务器的会话的标识符或由网站使用的其它数据。然而,对于使用 cookie,存在问题和缺点。例如,cookie 是特定于单一域或 web 页面的。换句话说,每两个来自不同域的服务器将需要不同的 cookie,即使该两个服务器提供可以共享公共登录证书的相关服务(例如,由位于不同域的公共服务提供商提供的日历和电子邮件服务)。相应地,用户为访问所述相关网站可能不得不输入冗余信息。进一步地,存储和管理随着用户访问更多网站而始终增长的 cookie 列表需要更多存储器空间和资源。在某些情况下,cookie 可以降低系统安全性,因为其可以被入侵者用于攻击系统或盗取用户数据。现在,对于许多用户和甚至网络运营商来说禁用或拦截 cookie 是常见的,使得客户端持久化或高速缓存的促进对许多用户不可用。

[0022] 此外,基于脚本语言(例如 JavaScript)的客户端持久化或高速缓存已通常限于提供存储在全局变量中的静态持久化信息,所述全局变量不能被改变或删除。该限制主要从以下情况下发生:当脚本文件首次在用户设备处被高速缓存(例如由用户的浏览器应用)时,脚本文件保持被高速缓存,并且通常不能使用动态信息(例如持久化或状态信息)来更新。换句话说,在将脚本文件用于提供客户端持久化或高速缓存的传统使用中,一旦嵌入在 web 内容中的脚本首次被加载并与伴随的 web 内容一起被高速缓存,则该文件不能被更新为反映关于该 web 内容的使用活动。相应地,任何所述持久化或其它被高速缓存的信息将不得不在用户与内容进行交互之前是提前已知的并被嵌入在内容中,由此限制了所述形式的脚本文件的可能应用。

[0023] 为解决该问题,图 1 的系统 100 引入了由服务器使用程序脚本提供客户端高速缓存的能力,其中,所述程序脚本是可在该脚本的初始访问和高速缓存之后动态更新的。特别地,系统 100 使能够区分对包括同一脚本文件的 web 内容的两个或更多请求。在首次访问脚本时,系统 100 可以请求会话配置信息。然后在后续访问时,系统 100 可以将会话配置信息并入脚本以便进行在客户设备处的高速缓存和使用。通过使用如此处描述的方法中引入的用于生成脚本的两步骤过程,系统 100 克服了脚本文件当用于提供客户端高速缓存时的静态内容限制。基于脚本的客户端持久化或高速缓存还有优势地克服了传统 cookie 的限制,所述限制例如是有限的数据能力(例如由于 cookie 的较小尺寸)、可应用于仅一个域(例如,由 cookie 提供的持久化是有限的并且受约束于安全漏洞)。

[0024] 如图 1 中所示,系统 100 包括用户设备(UE)101,该用户设备 101 具有经由通信网络 105 到一个或更多 web 服务器 103a-103n 的连接。在一个实施例中,UE 101 可以利用由服务器 103a-103n 中的一个或更多管理的客户端持久化或高速缓存来从网站访问信息或 web 内容。例如,UE 101 使用浏览器应用(例如会话客户端 111)向服务器 103a-103n 中的至少一个发送对 web 内容的请求。一旦接收到请求,则 web 服务器 103 生成包括脚本文件(例如 JavaScript 文件)的 web 内容代码(例如超文本标记语言(HTML)代码),其中,所述脚本文件用于存储 web 会话信息和提供客户端持久化。在一个实施例中,服务器 103 用于生成客户端持久化脚本文件的客户端高速缓存模块 107。脚本文件提供用于以下的逻辑:(1) 创建用于存储用户标识、认证或其它会话相关信息的会话变量;(2) 将会话变量存储在脚本文件中;以及然后(3) 在 UE 101 处(例如在 UE 101 的本地高速缓存器 113 中)高速缓存包含会话变量的脚本文件。已高速缓存的会话变量可以然后在将来请求同一 web 内容和/或会话时被从本地高速缓存器 113 中检索出,并被用于提供客户端持久化。

[0025] 在一个实施例中,当 UE 101 的会话客户端 111 访问所请求 web 内容中所引用的脚本文件时,脚本文件导致客户端持久化模块 107 确定该请求是否是用于访问该内容的首次请求(即,客户端持久化文件已被创建之前的请求,或不存在任何之前的会话信息的请求)。如果该请求是首次请求,则客户端持久化模块 107 基于从 UE 101 或 UE 101 的用户收集的会话配置信息创建会话变量,将那些变量存储为会话变量,并且立即指示会话客户端 111 刷新包括伴随的脚本文件的 web 内容的提供(render)。在进行刷新时,客户端高速缓存模块 107 检测到会话客户端 111 正在第二次(或后续)请求对内容和脚本文件的访问。同时,刷新使能够清除 web 内容中提供的初始脚本文件,从而客户端高速缓存模块 107 可以高速缓存已更新版本的脚本文件(例如,被修改为包含会话配置变量的版本的脚本文件)。特别地,第二次请求的检测和区分导致例如客户端高速缓存模块 107a 完成脚本文件创建过程,其通过将会话变量保存在脚本文件中,将脚本文件标记为用于在 UE 101 处进行高速缓存,以及将脚本文件发送到 UE 101。

[0026] 应当指出,此处描述的过程取决于客户端高速缓存模块 107a 区分对于同一 web 内容的两次请求的能力。然而,多数 web 内容当前使用 HTTP 来递送,其中,HTTP 被一般地定义为无状态客户/服务器协议。因此,HTTP 服务器通常不能识别对于相同源的两个请求之间的任何差别。因此,使用 HTTP 区分所述请求是重大技术挑战。为解决该挑战,客户端高速缓存模块 107a 利用 UE 101 与服务器 103 之间支持连接重用的连接(例如持久连接)的可用性来创建和在脚本文件中高速缓存会话变量。当用在此处时,支持连接重用的连接是指

可以使用同一连接处理多个请求 / 响应事务的连接。例如,HTTP 的当前版本(例如 1.1)引入 KeepAlive 连接作为支持连接重用的一种类型的连接。这些类型的连接可以也被称为持久连接。因为多个请求可以在同一连接上被处理,KeepAlive 连接的本质使客户端高速缓存模块 107a 能够在请求在同一连接(例如持久连接)上被接收时确定对同一 web 资源的请求的相对顺序。这样,客户端高速缓存模块 107a 可以顺序地生成会话变量,将其并入脚本文件(即更新脚本文件),以及在 UE101 处高速缓存已更新的脚本文件。

[0027] 应当指出,尽管关于 HTTP 1.1KeepAlive 连接描述了系统 100 的各种实施例,但应当设想,此处描述的方法适用于任何持久连接、支持 web 服务器与客户端之间的连接重用的其它类型连接,或者用于确定对同一 web 内容或资源的两个请求的相对顺序的机制。例如,在特定实施例中,系统 100 可以利用将 HTTP 与传输层安全(TLS) / 安全套接字层(SSL)合并的超文本传输协议安全(HTTPS)来提供加密和服务器的安全标识。特别地,HTTPS 在此处描述的方法中的使用使系统 100 即使客户端在服务器上产生了新套接字连接也能够使用关联于每个 HTTPS 连接的会话标识符(会话 ID)来关联客户端与服务器对。此在多个套接字连接上关联客户端与服务器对的能力使系统 100 能够为已配对客户端与服务器之间的多个请求重用连接。连接的重用接着使系统 100 能够确定请求的相对顺序或序列。

[0028] 会话客户端 111 可以然后使用已高速缓存的脚本文件和会话变量来提供所请求的 web 内容,并且在多个 web 会话之间提供客户端持久化。例如,UE 101 的用户可以关闭会话客户端 111,重启其,并且再次请求同一会话。在此情况下,UE 101 已在本地高速缓存器 113 中具有了会话变量(例如在已高速缓存的脚本文件内),并且因此,用户将不被客户端高速缓存模块 107 提示提供配置数据,除非用户选择重载会话或提前使会话数据被从本地高速缓存器 113 中移除。

[0029] 作为示例,系统 100 的通信网络 105 包括一个或更多网络,所述网络例如是数据网络(未示出)、无线网络(未示出)、电话网络(未示出)或其任意组合。应当设想,数据网络可以是任意局域网(LAN)、城域网(MAN)、广域网(WAN)、公共数据网络(例如互联网)或任意其它合适的分组交换网络,所述分组交换网络例如是商业拥有的私有分组交换网络,例如私有电缆或光纤网络。另外,无线网络可以是例如蜂窝网络,并且可以利用包括增强数据速率的全球演进(EDGE)、通用分组无线电业务(GPRS)、全球移动通信系统(GSM)、互联网协议多媒体子系统(IMS)、通用移动电信系统(UMTS)等的各种技术,以及例如微波接入全球互连(WiMAX)、长期演进(LTE)网络、码分复用(CDMA)、宽带码分复用(WCDMA)、无线高保真(WiFi)、卫星、移动自组网络(MANET)等的任意其它合适的无线介质。

[0030] UE 101 是任意类型的移动终端、固定终端或便携式终端,包括移动手机、站、单元、设备、多媒体计算机、多媒体图形输入板、互联网节点、发报机、桌面计算机、膝上电脑、个人数字助理(PDA)或其任意组合。还设想,UE 101 可以支持任意类型的与用户的界面(例如“可穿戴”电路等)。

[0031] 作为示例,UE 101 和客户端高速缓存模块 107a-107n 使用已熟知的、新的或仍在开发中的协议相互或与通信网络 105 的其它部件进行通信。在该上下文中,协议包括一组规则,所述规则定义通信网络 105 内的网络节点如何基于在通信链路上发送的信息相互进行交互。协议在每个节点内在不同运转层上起作用,从生成和接收各种类型的物理信号,到选择用于传输那些信号的链路,到对由那些信号指示的信息的格式化,到识别运行在计算

机系统上的哪个软件应用发送或接收该信息。用于在网络上交换信息的协议的概念上的不同层在开放系统互连(OSI)参考模型中描述。

[0032] 网络节点之间的通信通常通过交换离散的数据分组来实现。每个分组通常包括(1)关联于特定协议的报头信息,以及(2)净荷信息,所述净荷信息跟随在报头信息之后,并且包含可以独立于该特定协议而被处理的信息。在某些协议中,分组包括(3)尾部信息,该尾部信息跟随在净荷之后,并且指示净荷的结束。报头包括例如分组的源、其目的地、净荷的长度以及被协议使用的其它属性的信息。通常,用于特定协议的净荷中的数据包括用于不同协议的报头和净荷,其中,所述不同协议与OSI参考模型的不同的、较高层相关联。用于特定协议的报头通常指示被包含在其净荷中的下一个协议的类型。可以说较高层协议被封装在较低层协议中。被包括在穿过例如互联网的多个异构网络的分组中的报头通常包括如由OSI参考模型定义的物理(1层)报头、数据链路(2层)报头、互联网(3层)报头和传输(4层)报头以及各种应用报头(5层、6层和7层)。

[0033] 图2是根据一个实施例的客户端高速缓存模块的部件的图。作为示例,客户端高速缓存模块107包括一个或更多用于使用脚本文件提供客户端持久化的部件。应当设想,这些部件的功能可以被合并到一个或更多部件中,或者可以由具有等价功能的其它部件来实施。在该实施例中,客户端高速缓存模块107包括至少一个处理器201或其它控制逻辑,所述处理器201或其它控制逻辑用于执行至少一个用于实施客户端高速缓存模块107的功能的算法。例如,处理器201与连接检测器203进行交互,以便识别对于同一web内容或会话的特定一个或多个请求是否是经由支持连接重用的连接(例如KeepAlive连接或其它持久连接)接收的。在一个实施例中,连接检测器203可以访问会话客户端111用于与web服务器103进行通信的套接字(例如传输控制协议(TCP)套接字)。连接检测器203可以获取关联于套接字的标识符(例如套接字标识符(ID)),以便将对应连接识别为支持连接重用的连接(例如持久连接)。另外或可替换地,web服务器103可以使用不同的工作进程/线程来处理对于web资源的每个新到来请求。工作进程/线程然后在同一连接中处理客户端与web服务器103之间的进一步请求。由此,连接检测器203可以确定关联于合适的工作进程/线程的标识符,以便识别持久或可重用连接。

[0034] 在识别连接之后,如关于图1描述的那样,处理器201与脚本模块205进行交互以便生成可高速缓存的会话变量。特别地,脚本模块205确定客户端持久化脚本何时被首次高速缓存在UE101处(例如,何时不存在之前保存的会话数据或会话数据已被清除)。如关于图1描述的那样,使用支持连接重用的连接来实施所述检测。如果确定脚本文件首次被高速缓存,则脚本模块205与数据收集模块207进行交互,以便从UE101或UE101的用户收集会话配置信息。在一个实施例中,数据收集模块207发起向UE101的用于收集信息的表单的发送。当接收到会话配置信息时,数据收集模块207可以将该信息作为会话变量存储在会话数据库109中以便进行处理。然后,当脚本模块205检测到对脚本文件的第二次或后续请求时,脚本模块205可以将会话变量并入脚本文件,并且将文件和变量标记为用于在UE101处进行高速缓存。

[0035] 图3是根据一个实施例的用于提供客户端高速缓存的过程的流程图。在一个实施例中,客户端高速缓存模块107实施过程300,并且在例如图9中所示的包括处理器和存储器的芯片组中被实现。在步骤301中,客户端高速缓存模块107从UE101接收到对于访问

web 内容的请求。该请求可以包括对于会话脚本的请求。作为对该请求的响应,客户端高速缓存模块 107 确定该请求是否是在 UE 101 与 web 服务器 103 之间的支持连接重用的连接上接收的(步骤 303)。在一个实施例中,模块 107 可以验证用于进行通信的 HTTP 版本以及其是否支持所述连接(例如 KeepAlive 连接或其它持久连接)。如果请求不是在持久或可重用连接上接收的,则过程 300 结束。如果请求是在持久或可重用连接上接收的,则客户端高速缓存模块 107 确定该脚本文件请求(以及后续的高速缓存)是否是首次请求(步骤 305)。如之前描述的,为区分首次请求与后续请求,服务器可以识别不同 KeepAlive 连接,所述识别是通过由服务器进行的用于识别 KeepAlive 连接的几种方法中的任一个。例如,由于客户端高速缓存模块 107 可以访问被 web 服务器 103 和 UE 101 的会话客户端 111 用于进行通信的 TCP 套接字,所以套接字 ID 可以被用于区分 KeepAlive 会话。此外,web 服务器 103 可以使用不同的工作进程 / 线程来处理每个新的到来请求。通常,特定工作进程 / 线程将也在同一 KeepAlive 连接中处理进一步的请求。因此,客户端高速缓存模块 107 可以使用工作进程 / 线程 ID 来识别不同的 KeepAlive 会话。

[0036] 如果请求是首次请求,则客户端高速缓存模块 107 将脚本标记为被请求了一次,从而,下一次其被请求时,客户端高速缓存模块 107 知道该脚本之前已被请求了一次,并且应当使用会话信息(如果可用的话)来更新。由于这是首次请求,所以客户端高速缓存模块 107 生成用于在主动请求的 UE 与被请求的 web 内容之间进行通信的会话。为创建会话,如在步骤 309 中所示的那样,客户端高速缓存模块从主动请求的 UE 101 收集会话配置信息。配置信息可以包括用户标识和认证信息、UE 101 规格、被请求的网站地址、任何可能影响用户对被请求网站内容的访问的规则等。会话配置信息的某些部分可以从设备规格自动地收集,然而,例如用户标识或登录证书的信息的剩余部分从 UE 101 的用户来收集。为收集用户相关的信息,客户端高速缓存模块 107 向用户提供用于输入会话配置信息的界面。在收集会话配置信息(例如,或者自动地从 UE 101,或者手工地从用户)之后,客户端高速缓存模块 107 使用收集的信息指定会话变量(步骤 311)。会话变量被包括在更新的会话脚本文件中,其中,所述更新的会话脚本文件将在检测到对访问该脚本文件的第二个请求时被发送到 UE 101。相应地,客户端高速缓存模块 107 发起第二个请求,其是通过例如指示会话客户端 111 刷新或重载被请求的 web 内容(步骤 313)。

[0037] 由于刷新和重载,当客户端高速缓存模块 107 返回到步骤 305 以确定对于 web 内容和脚本文件的请求是否是首个请求时,模块 107 确定该请求是第二个请求,并且不是首个请求。这意味着,用于被请求会话的会话变量已被指定。换句话说,客户端高速缓存模块 107 已完成了如关于步骤 307 到 313 所描述的用于生成会话变量的过程。从在步骤 307 中标记的脚本得知这是第二个请求,客户端高速缓存模块 107 把在步骤 311 中指定的会话变量标记为可高速缓存的,从而 UE 101 能够将变量存储在其本地高速缓存 113 中(步骤 315)。在步骤 317 中,客户端高速缓存模块 107 将会话变量发送到 UE 101。例如,会话变量被包括在会话脚本文件中,以及,脚本文件与 web 内容一起被发送到 UE 101。换句话说,因为 web 内容包括嵌入的脚本文件,所以对脚本文件的第二个请求导致更新的脚本文件(例如包括会话变量的脚本文件)的发送,以便在 UE 101 处进行高速缓存。

[0038] 在另一实施例中,客户端高速缓存模块 107 可以限制会话变量的范围,以便防止入侵者获取对会话变量的访问,所述获取是通过例如邀请用户访问恶意网站、在网站中放

置对原始服务器上的会话的引用、将浏览器设置改变为代替从服务器而从本地高速缓存加载会话，或将浏览器设置为将会话变量发回恶意站点。作为示例，模块 107 可以通过代替全局变量而使用本地变量作为会话变量来限制变量的范围。此外，模块 107 可以对被允许访问变量的网站施加限制。表 1 的样例代码是所述范围限制的一种实现。

[0039] 表 1

[0040]

```
function get_private_session() {
    var private_session_id = "ID setup by User and Server";
    if (window.location.host != "Server hostname or any other allowed websites") {
        return 0;
    } else {
        return private_session_id;
    }
}
```

[0041] 表 1 中的 window.location.host 是保留变量，其识别当前页面被下载自的网站的主机名。因此，通过使用该技术，会话变量的范围被灵活地设置。例如，网站“sample1.com”可以建立会话变量并允许“sample2.com”使用该变量。相反，传统 cookie 对会话变量有严格限制，从而 cookie 会话变量可以仅在该 cookie 被建立所在的初始 web 服务器的范围内被使用。

[0042] 图 4 是根据一个实施例的、用户从服务器请求 web 内容的过程的流程图。在一个实施例中，UE 101 的会话客户端 111 实施过程 400，并且可以在例如图 9 中所示的包括处理器和存储器的芯片组中被实现。在一个实施例中，UE 101 激活会话客户端 111，以便获得使用客户端持久化或高速缓存对 web 内容的访问。在步骤 401 中，会话客户端 111 生成对于 UE 101 与 web 服务器 103 之间用于所述内容的通信建立请求，并且将该请求发动到服务器 103。因为 web 内容包括客户端持久化，所以会话请求还包括对可以提供客户端持久化的脚本文件的请求。在步骤 403 中，会话客户端 111 从服务器接收到对于提供与会话建立相关的信息（例如用户 ID、密码等）的请求。在一个实施例中，服务器数据收集请求可以包括用于手工输入会话配置信息的表单。另外或可替换地，服务器 103（例如经由客户端高速缓存模块 107）可以直接从 UE 101（例如从本地高速缓存器 113）读取会话配置信息。如果用户信息被服务器经过步骤 407 认可（例如信息包括任何用于检索内容的认证信息），则会话客户端 111 从服务器接收会话发起允许（步骤 409）。作为示例，所述允许可以包括用于如关于图 3 的步骤 313 描述的那样发起对脚本文件的第二个请求的命令或消息（例如刷新命令）。当接收到允许和/或刷新命令时，会话客户端 111 使用包括在允许消息中的信息在步骤 411 中向服务器发送对脚本文件的第二个请求。在步骤 413 中，会话客户端 111 接收到所请求的脚本文件，该脚本文件包括按照过图 3 的步骤 317 生成的会话变量。由于服务器已将会话变量标记为可在客户端高速缓存，所以会话客户端 111 在本地高速缓存器 113 中刷新并存储会话变量（步骤 415）。在步骤 417 中，会话客户端 111 通过由服务器生成的会话并且使用来自 UE 101 的本地高速缓存器 113 的会话变量与网站进行通信。

[0043] 在另一实施例中，会话客户端 111 可以向服务器 103 发送这样的请求，所述请求已被使用持久化或其它已高速缓存的信息进行了调整。换句话说，会话客户端 111 基于存储在本地高速缓存器 113 中的会话变量修改内容请求，使得该请求将特别要求提供之前会话

的继续的内容。服务器 103 然后返回仅满足已调整请求的内容。另外,会话客户端 111 可以在对来自服务器 103 的内容的请求中包括认证报头。该认证从客户端高速缓存中找到,并且服务器 103 返回所请求的内容。

[0044] 图 5 是根据一个实施例的过程的流程图,该过程用于向设备提供用于 web 内容的后续请求的客户端高速缓存。如图 3 中可见的,首次 UE 101 请求特定网站的 web 内容时,UE 101 与网站之间的通信会话(例如 web 会话)由对应的 web 服务器 103 生成。作为创建该 web 会话的一部分,如之前描述的那样,web 服务器 103 的客户端高速缓存模块 107 生成并在 UE101 的本地高速缓存器 113 中高速缓存会话变量。因此,对于 UE 101 的会话客户端 111 产生的对同一 web 内容或网站的任何后续请求,会话变量可以由会话客户端 111 从本地高速缓存器 113 中检索出。相应地,在步骤 501 中,服务器 103 从 UE 接收到对 web 内容的请求,其中,该 web 内容之前已被同一 UE 101 经由会话客户端 111 请求过。web 服务器向会话客户端 111 返回所请求的 web 内容。作为响应,会话客户端通过从本地高速缓存器 113 检索参考客户端持久化脚本来提供 web 内容。特别地,会话客户端 111 从已高速缓存的脚本文件中提取会话变量,并且应用会话变量以提供和呈现内容。这样,内容可以由之前 web 会话确定的状态被呈现。

[0045] 图 6 是时序图,示出了根据一个实施例的用于提供客户端持久化的消息序列和过程。网络进程用竖线来代表。从一个进程传递到另一个的消息用水平箭头来代表。由进程实施的步骤用文本来指示。图 6 中所代表的进程是 UE 101 的会话客户端 111 和 web 服务器 103。图 6 的示例讨论用于使用 JavaScript 和 HTTP 提供客户端持久化的过程 600。

[0046] 在 605 处,会话客户端 111 请求包括客户端持久化支持的 web 服务器 103 上的网页。web 服务器 103 返回网页作为“session.php”,其中,“session.php”进一步引用客户端持久化脚本文件“session-js.php”。当接收到网页时,会话客户端 111 产生用于提供 session.php 网页的对 session-js.php 脚本的请求(在 607 处)。在本示例中,web 服务器 103 确定该请求是会话客户端 111 首次请求 session-js.php 脚本(在 609 处),并且在 session-js.php 脚本中为会话客户端 111 或会话客户端 111 的用户生成用于生成新会话的表单(在 611 处)。

[0047] 在 613 处,会话客户端 111 的用户填充表单,以及,会话客户端 111 将表单结果发送到 web 服务器 103 上的单独网页“interim.php”。当接收到 interim.php 网页时,web 服务器 103 根据表单中提供的信息准备会话信息(在 617 处)。同时,web 服务器 103 将浏览器重定向回 session.php 网页,以便在会话客户端 111 处刷新网页和对应脚本(在 619 处)。

[0048] 作为重定向的响应,会话客户端 111 产生向 web 服务器 103 对 session.php 网页的另一请求(在 621 处)。当提供 session.php 网页时,会话客户端 111 还基于来自 web 服务器 103 的重定向或刷新命令产生对在 session.php 网页中所引用的 session-js.php 脚本的另一请求(在 623 处)。web 服务器 103 接收到已重定向的对 session-js.php 脚本的请求,并且确定该已重定向的请求是对于 session-js.php 脚本的第二个请求(在 625 处)。相应地,web 服务器 103 从在 617 处收集的信息创建会话变量,并且在 session-js.php 脚本中返回会话变量(在 627 处)。在发送更新的脚本之前,web 服务器 103 还将会话变量标记为可在客户端处(例如在 UE 101 处)高速缓存。更新的 session-js.php 然后被会话客户端 111 用于基于会话变量完成 session.php 网页的提供。

[0049] 在一个实施例中, JavaScript 可以当作与 web 内容相关联的脚本文件的语言被使用。表 2 是使用 JavaScript 并且关于图 6 的过程被描述的会话的样例实现。

[0050] 表 2 :样例实现

[0051]

```
|||||||||||||||||||||| COMPONENT 1 ||||||||||||||||||||  
y3fu@ibex:/var/www$ cat globals.php  
<?php  
$g_server_hostname = "localhost";  
$g_session_file_folder = "/tmp";  
?>  
  
|||||||||||||||||| COMPONENT 2 ||||||||||||||||||||  
y3fu@ibex:/var/www$ cat session-js.php  
<?php  
    include "globals.php";  
    $pid = getmypid();  
    $filename = "$g_session_file_folder/$pid.dat";  
    if (file_exists($filename)) {  
        header("Cache-Control: max-age=3600");  
        $fp = fopen($filename, "r");  
        $line = fgets($fp);  
        while ($line) {  
            echo $line;  
            $line = fgets($fp);  
        }  
        fclose($fp);  
        unlink($filename);  
    }  
    function foo() {  
    }  
<?php  
    } else {  
?>  
    function login() {  
        var location = "http://<?php echo $g_server_hostname; ?>/interim.php?session=" +  
        document.getElementById('session').value;  
        window.location=location;  
    }  
}
```

[0052]

```

function foo() {
    var myHTML = "<input id='session'><input type='button' value='OK' onclick='login()'>";
    document.getElementById('uiBody').innerHTML = myHTML;
}
<?php
}
?>

////////////////// COMPONENT 3 ///////////////////
y3fu@ibex:/var/www$ cat interim.php
<?php
    require "globals.php";
    $session = htmlspecialchars($_GET["session"]);
    $pid = posix_getpid();
    $filename = "$g_session_file_folder/$pid.dat";
    if (file_exists($filename)) {
        echo "fatal error";
    } else {
        $fp = fopen($filename, "w");
        $line = "var session_id=$session;\n";
        fwrite($fp, $line);
        $line =
    }
    function get_private_session() {
        var private_session_id = session_id + 1;
        if (window.location.host != '$g_server_hostname') {
            return 0;
        } else {
            return private_session_id;
        }
    };
    fwrite($fp, $line);
    fclose($fp);
    header("Location: http://$g_server_hostname/session.php");
}
?>

////////////////// COMPONENT 4 ///////////////////
y3fu@ibex:/var/www$ cat session.php
<html>
<head>
<script language="JavaScript" src="session-js.php"></script>
<script type="text/javascript" language="JavaScript">

function print_session() {
    if (session_id) {
        alert("session is: " + session_id + ", private session is: " + get_private_session());
    } else {
        alert("no session setup yet");
    }
}
</script>
</head>
<body onload="foo()">
<span id="uiBody"></span>
<input type="button" value="Check Session" onclick="print_session()">
</body>
</html>

```

[0053] 表 2 的第一个组件 (COMPONENT1) 是初始化组件, 在其中定义了会话变量的主机名和位置。表 2 的组件 2 (COMPONENT2) 是如在图 3 中讨论的脚本文件。回头参考图 6 的过

程 605, 会话客户端 111 使用 GET 命令向 web 服务器 103 请求, 并且要求 session.php 网页, 其中, 该 session.php 网页是表 2 中的组件 4。

[0054] 如在表 2 的组件 4 (COMPONENT4) 中可见的, session.php 网页如在图 6 的过程 607 中反映的那样引用 session.js.php 脚本。session.js.php 脚本被示为表 2 的组件 2。脚本 session.js.php 被服务器执行使得服务器 103 如根据图 6 的过程 609 那样检查这是否是对该文件脚本的第一个请求并创建新会话。在 611 处, 服务器向 UE 发送对会话配置所需的用户数据的请求。在 613 处, 用户将所请求的信息输入由服务器 103 提供的界面中, 并且根据过程 615, 会话客户端 111 通过 interim.php 将该信息发送给服务器 103, 其中, interim.php 是表 2 的组件 3 (COMPONENT3)。在图 6 的 617 处, 服务器 103 基于通过 interim.php 从用户接收的配置信息准备脚本文件(例如 session.js.php)的内容, 以及在 619 处, 服务器 103 第二次将用户定向到 session.php。过程 621 和 623 是过程 605 和 607 的第二次重复, 其中, session.php 的提供将调用 session.js.php。然而与 609 处不同, 服务器 103 知道这是对 session.js.php 的第二个请求(在 625 处)。因此, 服务器 103 将在过程 617 中准备的会话变量标记为可高速缓存, 并根据过程 627 将所述变量发送给会话客户端 111。

[0055] 所描述的方法未对会话变量施加任何限制, 这意味着变量可以甚至是编程语言(例如 JavaScript)的函数。把会话变量建立在脚本语言的基础上有利地使脚本文件能够支持更多样和更高质量的客户端持久化功能。相反, 如之前指出的, cookie 是较小有限尺寸的字符串, 其功能容易和例行地被限制, 或否则受限于规范、运营商、用户、应用(例如浏览器)等。此外, 在所描述的方法中, 会话变量可以被灵活地配置为被一个或更多网站使用, 而 cookie 通常可以仅在其最初被设置所在的网站上使用。所引入的过程还有利地使客户端持久化脚本能够被动态地更新, 以便反映变化的会话变量和情况。

[0056] 图 7 是根据一个实施例的用于提供与客户端高速缓存相关的数据的示例性用户界面的图。如之前描述的, 为在 UE 101 与网站之间生成对话, web 服务器 103 可以从用户请求会话配置信息。在此情况下, 服务器 103 可以创建表单并将其发送到 UE 101 的会话客户端 111, 以便用户和 / 或 UE 101 填完整并将其返回给服务器 103。图 7 是请求所述信息的样例用户界面。在本示例中, 用户和 / 或 UE 101 将输入用于认证目的的用户 ID 和密码。取决于所请求的 web 内容, 服务器 103 可以还要求其它信息, 例如用户的邮政编码、年龄和使用下拉菜单的兴趣类别, 以便能够为用户提供更好或定制的服务。

[0057] 此处描述的用于提供客户端高速缓存的过程可以经由软件、硬件(例如通用处理器、数字信号处理(DSP)芯片、专用集成电路(ASIC)、现场可编程门阵列(FPGA)等)、固件或其组合来有利地实现。下面详述用于实施所描述的功能的所述示例性应用。

[0058] 图 8 示出了计算机系统 800, 其中, 本发明的实施例可以在其上被实现。尽管关于特定设备或装置示出了计算机系统 800, 但应当设想, 在图 8 内, 其它设备或装置(例如网络单元、服务器等)可以部署所示的系统 800 的硬件和部件。计算机系统 800 被编程为(例如经由计算机程序代码或指令)如此处描述的那样提供客户端高速缓存, 并且包括例如总线 810 的通信设备, 所述通信设备用于在计算机系统 800 的其它内部和外部部件之间传递信息。信息(也称为数据)被表示为可测量现象的物理表达, 所述可测量现象通常是电压, 但在其它实施例中包括例如磁、电磁、压力、化学、生物、分子、原子、次原子、量子作用的现象。例如, 例如北和南磁场、或零和非零电压代表二进制数字(位)的两个状态(0, 1)。其它现象可

以代表更高基数的数字。测量前对多个同时定量状态的叠加代表定量位(qubit)。一个或更多数字的序列构成数字数据，所述数字数据被用于代表数量或字符的代码。在某些实施例中，称为模拟数据的信息用特定范围内的接近连续的可测量值来表示。计算机系统 800 或其部分构成这样的装置，所述装置用于实施提供客户端高速缓存的一个或更多步骤。

[0059] 总线 810 包括一个或更多并行的信息导线，从而信息可以快速地在耦合到总线 810 的设备之间传输。用于处理信息的一个或更多处理器 802 与总线 810 相耦合。

[0060] 处理器 802 如由计算机程序代码指定的那样对信息实施一组操作，其中，所述计算机程序代码与提供客户端高速缓存相关。计算机程序代码是用于将处理器和 / 或计算机系统运转为实施指定功能的一组指令或提供指令的语句。代码例如可以用被编译为处理器的原生指令集的计算机编程语言来书写。代码可以还直接使用原生指令集(例如机器语言)来书写。所述一组操作包括从总线 810 取入信息以及将信息放到总线 810 上。所述一组操作通常还包括比较信息的两个或更多单元、移动信息的单元的位置以及合并信息的两个或更多单元，其中，所述合并例如是通过加或乘或例如或、异或(XOR)和与的逻辑运算。可以被处理器实施的所述一组操作中的每个操作通过称为指令的信息向处理器表示，所述指令例如是一个或更多数字的运算码。例如运算码序列的将由处理器 802 执行的操作序列构成处理器指令，所述处理器指令也称为计算机系统指令，或简单地称为计算机指令。处理器可以特别单独或组合地被实现为机械、电、磁、光、化学或量子部件。

[0061] 计算机系统 800 还包括耦合到总线 810 的存储器 804。例如随机访问存储器(RAM)或其它动态存储设备的存储器 804 存储包括处理器指令的信息，所述信息用于提供客户端高速缓存。动态存储器允许存储在其中的信息被计算机系统 800 改变。RAM 允许存储在称为存储器地址的位置处的信息单元独立于相邻地址处的信息而被存储和检索。存储器 804 还被处理器 802 用于存储执行处理器指令期间的临时值。计算机系统 800 还包括耦合到总线 810 的用于存储包括指令的静态信息的只读存储器(ROM) 806 或其它静态存储设备，其中，所述静态信息不被计算机系统 800 改变。某些存储器由易失性存储器组成，所述易失性存储器在失去供电时丢失存储在其中的信息。非易失性(持久)存储器设备 808 也耦合到总线 810，其例如是磁盘、光盘或闪存卡，用于存储包括指令的信息，所述信息甚至在计算机系统 800 被关闭或否则失去供电时也存留。

[0062] 从例如包含被人类用户操作的字母数字键的键盘或传感器的外部输入设备 812 将用于提供客户端高速缓存的包括指令的信息提供给总线 810，以便被处理器使用。传感器检测其附近的状况，并且将那些检测变换为与用于在计算机系统 800 中表示信息的可测量现象相兼容的物理表达。主要用于与人类进行交互的耦合到总线 810 的其它外部设备包括：显示设备 814，其例如是阴极射线管(CRT)或液晶显示器(LCD)、或等离子屏幕或打印机，用于呈现文本或图像；以及指点设备 816，其例如是鼠标或轨迹球或光标方向键、或运动传感器，用于控制呈现在显示器 814 上的小光标图像的位置，以及发出与呈现在显示器 814 上的图形元素相关联的命令。在一些实施例中，例如，在其中计算机系统 800 在没有人类输入的情况下自动实施全部功能的实施例中，省略外部输入设备 812、显示设备 814 和指点设备 816 中的一个或更多。

[0063] 在所示的实施例中，例如专用集成电路(ASIC)820 的专用硬件耦合到总线 810。专用硬件被配置为实施这样的操作，所述操作不被处理器 802 对于专门目的足够快地实施。

专用 IC 的示例包括用于为显示器 814 生成图形的图形加速器卡、用于对在网络上发送的消息进行加密和解密的密码板、语音识别以及与例如机械臂和医学扫描装置的特殊外部设备的接口，其用于重复实施某些复杂的操作序列，所述复杂操作序列在硬件中被更高效地实现。

[0064] 计算机系统 800 还包括一个或更多耦合到总线 810 的通信接口 870 的实例。通信接口 870 提供与使用其自己的处理器运转的多种外部设备的单向或双向通信耦合，所述外部设备例如是打印机、扫描仪和外部磁盘。通常，耦合是通过连接到本地网络 880 的网络链路 878，其中，具有其自己的处理器的多种外部设备连接到本地网络 880。例如，通信接口 870 可以是个人计算机上的并行端口或串行端口或通用串行总线(USB)端口。在某些实施例中，通信接口 870 是为对应类型电话线路提供信息通信连接的综合服务数字网络(ISDN)卡或数字用户线路(DSL)卡或电话调制解调器。在某些实施例中，通信接口 870 是电缆调制解调器，该电缆调制解调器将总线 810 上的信号转换为用于同轴电缆上的通信连接的信号，或转换为用于光缆上的通信连接的光信号。作为另一示例，通信接口 870 可以是局域网(LAN)卡，该局域网卡用于提供与例如以太网的兼容 LAN 的通信连接。也可以实现无线链路。对于无线链路，通信接口 870 发送或接收或发送并接收携带例如数字数据的信息流的电、声或电磁信号，所述电磁信号包括红外和光信号。例如，在例如像蜂窝电话的移动电话的无线手持设备中，通信接口 870 包括称为无线电收发器的无线电频带电磁发送器和接收器。在特定实施例中，通信接口 870 使能连接到用于向 UE 101 提供客户端高速缓存的通信网络 105。

[0065] 术语“计算机可读介质”当用在此处时是指任意参与向处理器 802 提供信息的介质，所述信息包括用于执行的指令。所述介质可以采取许多形式，包括但不限于计算机可读存储介质(例如非易失性介质、易失性介质)和传输介质。例如非易失性介质的非瞬态介质例如包括例如存储设备 808 的光或磁盘。易失性介质例如包括动态存储器 804。传输介质例如包括同轴电缆、铜线、光缆以及在没有线或缆的情况下通过空间的载波，所述载波例如是声波和包括无线电、光和红外波的电磁波。信号包括通过传输介质传输的人造的幅度、频率、相位、极性或其它属性上的瞬时变化。常见形式的计算机可读介质例如包括软盘、软磁盘、硬盘、磁带、任何其它磁介质、CD-ROM、CDRW、DVD、任何其它光介质、打孔卡、纸带、光标记表、具有孔或其它光可识别标记的图案的任何其它物理介质、RAM、PROM、EPROM、FLASH-EPROM、任何其它存储器芯片或卡带、载波或计算机可以从其读取的任意其它介质。术语计算机可读存储介质此处被用于指除传输介质外的任意计算机可读介质。

[0066] 被编码在一个或更多现实介质中的逻辑包括计算机可读存储介质和例如 ASIC 820 的专用硬件上的处理器指令的一个或这两者。

[0067] 网络链路 878 通常使用传输介质通过一个或更多网络提供与其它设备的信息通信，其中，该其它设备使用或处理所述信息。例如，网络链路 878 可以通过本地网络 880 提供与主机计算机 882 或与由互联网服务提供商(ISP)运营的装置 884 的连接。ISP 装置 884 接着通过现在通常称为互联网 890 的网络的公共全球分组交换通信网络提供数据通信服务。

[0068] 连接到互联网的称为服务器主机 892 的计算机是这样的进程的宿主，该进程响应于在互联网上接收的信息而提供服务。例如，服务器主机 892 是这样的进程的宿主，该进程

提供代表视频数据的信息，该视频数据用于在显示器 814 处呈现。应当设想，系统 800 的部件可以以各种配置被部署在例如主机 882 和服务器 892 的其它计算机系统中。

[0069] 本发明的至少一些实施例涉及将计算机系统 800 用于实现此处描述的技术中的一些或全部。根据本发明的一个实施例，那些技术被计算机系统 800 响应于处理器执行包含在存储器 804 中的一个或更多一个或更多处理器指令的序列而实施。所述也称为计算机指令、软件和程序代码的指令可以被从例如存储设备 808 或网络链路 878 的另一计算机可读介质读入存储器 804 中。包含在存储器 804 中的指令序列的执行使得处理器 802 实施此处描述的方法步骤中的一个或更多。在可替换实施例中，例如 ASIC 820 的硬件可以代替或与软件结合地被用于实现本发明。由此，除非此处明确指出，否则本发明的实施例不限于硬件和软件的任意特定组合。

[0070] 通过通信接口 870 在网络链路 878 和其它网络上发送的信号携带去往和来自计算机系统 800 的信息。计算机系统 800 可以通过网络链路 878 和通信接口 870 特别通过网络 880、890 发送和接收包括程序代码的信息。在一个使用互联网 890 的示例中，服务器主机 892 通过互联网 890、ISP 装置 884、本地网络 880 和通信接口 870 发送用于特定应用的程序代码，其中，所述程序代码是从计算机 800 发送的消息所请求的。接收的代码可以在其被接收时被处理器 802 执行，或者可以被存储在存储器 804 或存储设备 808 或其它非易失性存储中以便稍后执行，或者这两者都被进行。这样，计算机系统 800 可以获得采用载波上的信号形式的应用程序代码。

[0071] 在将一个或更多指令或数据序列携带到处理器 802 以便执行时，可以涉及各种形式的计算机可读介质。例如，指令和数据可以初始被携带在例如主机 882 的远程计算机的磁盘上。远程计算机将指令和数据加载到其动态存储器中，并且使用调制解调器在电话线上发送指令和数据。计算机系统 800 本地的调制解调器在电话线上接收指令和数据，并且使用红外发送器将指令和数据转换为红外载波上的信号，其中，红外载波充当网络链路 878。充当通信接口 870 的红外检测器接收到携带在红外信号中的指令和数据，并且将代表指令和数据的信息放到总线 810 上。总线 810 将信息携带到存储器 804，其中，处理器 802 从存储器 804 检索并使用与指令一起被发送的数据中的一些执行指令。在存储器 804 中接收的指令和数据，在被处理器 802 执行之前或之后，可以可选地被存储在存储设备 808 中。

[0072] 图 9 示出了芯片组 900，其中，本发明的实施例可以在芯片组 900 上实现。芯片组 900 被编程为如此处描述的那样提供客户端高速缓存，并且例如包括被并入一个或更多物理包(例如芯片)中的、关于图 8 描述的处理器和存储器部件。作为示例，物理包包括一个或更多材料、部件和 / 或电线在结构化设备(例如底板)上的排列，所述结构化设备用于提供例如物理长度、尺寸保护和 / 或电作用限制的一个或更多特征。应当设想，在特定实施例中，芯片组可以在单一芯片中实现。芯片组 900 或其部分构成这样的装置，所述装置用于实施提供客户端高速缓存的一个或更多步骤。

[0073] 在一个实施例中，芯片组 900 包括例如总线 901 的通信设备，总线 901 用于在芯片组 900 的部件之间传递信息。处理器 903 具有到总线 901 的连接，其用于执行和处理存储在例如存储器 905 中的指令和信息。处理器 903 可以包括一个或更多处理核心，其中，每个核心被配置为独立地实施。多核处理器使能进行在单一物理包内的多处理。多核处理器的示例包括两个、四个、八个或更多数量的处理核心。可替换地或另外，处理器 903 可以包括

一个或更多微处理器，所述微处理器被经由总线 901 串联地配置为，使能进行指令、关联和多线程的独立执行。处理器 903 可以还附带用于实施特定处理功能和任务的一个或更多专用部件，例如一个或更多数字信号处理器(DSP) 907、或者一个或更多专用集成电路(ASIC) 909。DSP 907 通常被配置为，独立于处理器 903 实时地处理现实世界信号(例如声音)。类似地，ASIC 909 可以被配置为，实施不被通用处理器容易地实施的专业功能。用于帮助实施此处描述的发明功能的其它专业部件包括一个或更多现场可编程门阵列(FPGA)(未示出)、一个或更多控制器(未示出)或者一个或更多其它专用计算机芯片。

[0074] 处理器 903 和附属部件具有经由总线 901 到存储器 905 的连接。存储器 905 包括用于存储可执行指令的动态存储器(例如 RAM、磁盘、可写光盘等)和静态存储器(例如 ROM、CD-ROM 等)这两者，所述可执行指令当被执行时实施此处描述的发明步骤来提供客户端高速缓存。存储器 905 还存储与发明步骤的执行相关联或由发明步骤的执行生成的数据。

[0075] 图 10 是根据一个实施例的用于通信的移动终端(例如手机)的示例性部件的图，该移动终端能够在图 1 的系统中运转。在某些实施例中，移动终端 1000 或其部分构成这样的装置，所述装置用于实施提供客户端高速缓存的一个或更多步骤。一般地，通常在前端和后端特征方面定义无线电接收器。接收器的前端包括无线电频率(RF)电路的全部，而后端包括基带处理电路的全部。当用在本申请中时，术语“电路”是指这两者：(1)仅硬件实现(例如用仅模拟和 / 或数字电路的实现)，以及(2)电路和软件(和 / 或固件)的组合(例如，在适用的特定上下文中，是指包括数字信号处理器的处理器、软件和存储器的组合，所述处理器、软件和存储器一起运转使得例如移动电话或服务器的设备实施各种功能)。“电路”的该定义适用于此术语在本申请中包括任何权利要求中的全部使用。作为进一步的示例，当用在本申请中并且在适用的特定上下文中时，术语“电路”将还覆盖仅一个处理器(或多个处理器)及它的(或它们的)附属软件和 / 或固件的实现。在适用的特定上下文中，术语“电路”将还覆盖例如移动电话中的基带集成电路或应用处理器集成电路，或者蜂窝网络设备或其它网络设备中的类似集成电路。

[0076] 电话的相关内部部件包括主控制单元(MCU) 1003、数字信号处理器(DSP) 1005 以及包括麦克风增益控制单元和扬声器增益控制单元的接收器 / 发送器单元。主显示单元 1007 向用户提供显示，以便支持实施或支持提供客户端高速缓存的步骤的各种应用和应用终端功能。显示器 1007 包括显示器电路，该显示器电路被配置为，显示移动终端(例如移动电话)的用户界面的至少一部分。另外，显示器 1007 和显示器电路被配置为，促进用于对移动终端的至少一些功能进行控制。音频功能电路 1009 包括麦克风 1011 以及对来自麦克风 1011 的语音信号输出进行方法的麦克风放大器。从麦克风 1011 输出的放大的语音信号被馈入编 / 解码器(CODEC) 1013。

[0077] 无线电部分 1015 放大功率并且转换频率以便经由天线 1017 与包括在移动通信系统中的基站进行通信。如本领域中已知的那样，功率放大器(PA) 1019 和发送器 / 调制电路运用地响应于 MCU 1003，其中，来自 PA 1019 的输出耦合到双工器 1021 或环形器或天线开关。PA 1019 还耦合到电池接口和功率控制单元 1020。

[0078] 在使用中，移动终端 1001 的用户对麦克风 1011 说话，以及他的或她的语音与任何检测到的背景噪声一起被转换成模拟电压。模拟电压然后通过模拟到数字转换器(ADC) 1023 被转换成数字信号。控制单元 1003 将数字信号选路到 DSP 1005 中以便在其中进行

处理,所述处理例如是语音编码、信道编码、加密和间插。在一个实施例中,已处理的语音信号被未单独示出的单元进行编码,所述编码使用例如全球演进(EDGE)、通用分组无线电服务(GPRS)、全球移动通信系统(GSM)、互联网协议多媒体子系统(IMS)、通用移动电信系统(UMTS)等的蜂窝传输协议,以及例如微波接入(WiMAX)、长期演进(LTE)网络、码分复用(CDMA)、宽带码分复用(WCDMA)、无线高保真(WiFi)、卫星等的任意其它合适的无线介质。

[0079] 已编码的信号然后被选路到均衡器 1025,以便对在通过空气传输期间发生的例如相位和幅度失真的任何频率相关损坏进行补偿。在对位流进行均衡化之后,调制器 1027 将信号与在 RF 接口 1029 中生成的 RF 信号合并。作为频率或相位调制的示例,调制器 1027 生成正弦波。为准备用于传输的信号,向上转换器 1031 将从调制器 1027 输出的正弦波与由合波器 1033 生成的正弦波合并,以便达到期望的传输频率。信号然后通过 PA 1019 被发送,以便将信号增大到合适的功率水平。在实际系统中,PA 1019 充当可变增益放大器,其增益由 DSP 1005 根据从网络基站接收到的信息进行控制。信号然后在双工器 1021 中被进行滤波,并且可选地被发送到天线耦合器 1035,以便匹配用于提供最大功率传输的阻抗。最后,信号经由天线 1017 被发送到本地基站。可以提供自动增益控制(AGC)来控制接收器的最终阶段的增益。信号可以被从该处转发到远端电话,该远端电话可以是另一蜂窝电话、其它移动电话或连接到公共电话交换网(PSTN)或其它电话网络的陆地线路。

[0080] 发送到移动终端 1001 的语音信号经由天线 1017 被接收,并且立即被低噪声放大器(LNA)1037 放大。向下转换器 1039 降低载波频率,而解调器 1041 剥除 RF 留下仅数位流。信号然后经过均衡器 1025,并且被 DSP 1005 处理。全部在可以被实现为中央处理单元(CPU)(未示出)的主控制单元(MCU)1003 的控制下,数字到模拟转换器(DAC)1043 对信号进行转换,以及产生的输出通过扬声器 1045 被发送给用户。

[0081] MCU 1003 接收包括来自键盘 1047 的输入信号的各种信号。键盘 1047 和 / 或结合其它用户输入部件(例如麦克风 1011)的 MCU 1003 包括用于管理用户输入的用户界面电路。MCU 1003 运行用户界面软件来促进用户对移动终端 1001 用于提供客户端高速缓存的至少一些功能的控制。MCU 1003 还分别向显示器 1007 和语音输出开关控制器递送显示命令和开关命令。进一步地,MCU 1003 与 DSP 1005 交换信息,并且可以访问可选地包含的 SIM 卡 1049 和存储器 1051。另外,MCU 1003 执行终端所需的各种控制功能。取决于实现,DSP 1005 可以实施对语音信号的多种常规数字处理功能中的任一个。另外,DSP 1005 根据由麦克风 1011 检测到的信号确定本地环境的背景噪声水平,并且将麦克风 1011 的增益设置为这样的水平,该水平被选择用于对移动终端 1001 的用户的自然趋势进行补偿。

[0082] CODEC 1013 包括 ADC 1023 和 DAC 1043。存储器 1051 存储包括来电音数据的各种数据,并且能够存储包括例如经由全球互联网接收的音乐数据的其它数据。软件模块可以驻留在 RAM 存储器、闪存器、寄存器或本领域中已知的任意其它形式的可写存储介质中。存储器设备 1051 可以但不限于是单一存储器、CD、DVD、ROM、RAM、EEPROM、光存储器或能够存储数字数据的任意其它非易失性存储介质。

[0083] 可选地包含的 SIM 卡 1049 例如携带重要信息,所述重要信息例如是蜂窝电话号码、提供服务的载波、订阅详情和安全信息。SIM 卡 1049 主要作用为,在无线电网络上标识移动终端 1001。卡 1049 还包含存储器,该存储器用于存储个人电话号码簿、文本消息以及用户特定的移动终端设置。

[0084] 尽管已结合许多实施例和实现描述了本发明，但本发明并非如此受限，而是覆盖落在所附权利要求的范围内的各种显而易见的修改和等价安排。尽管在权利要求中以特定组合表述了本发明的特征，但应当设想，这些特征可以以任意组合和顺序来安排。

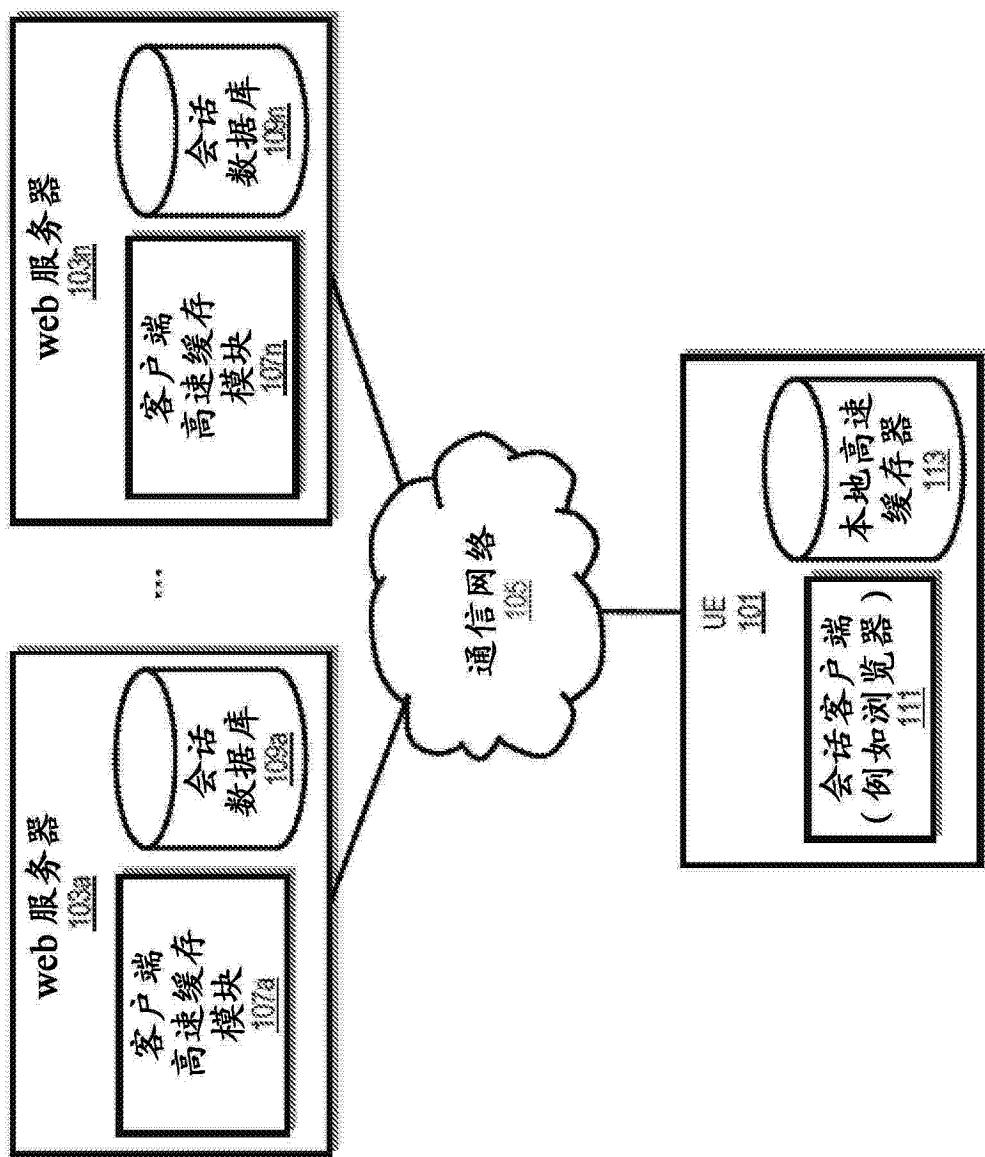


图 1

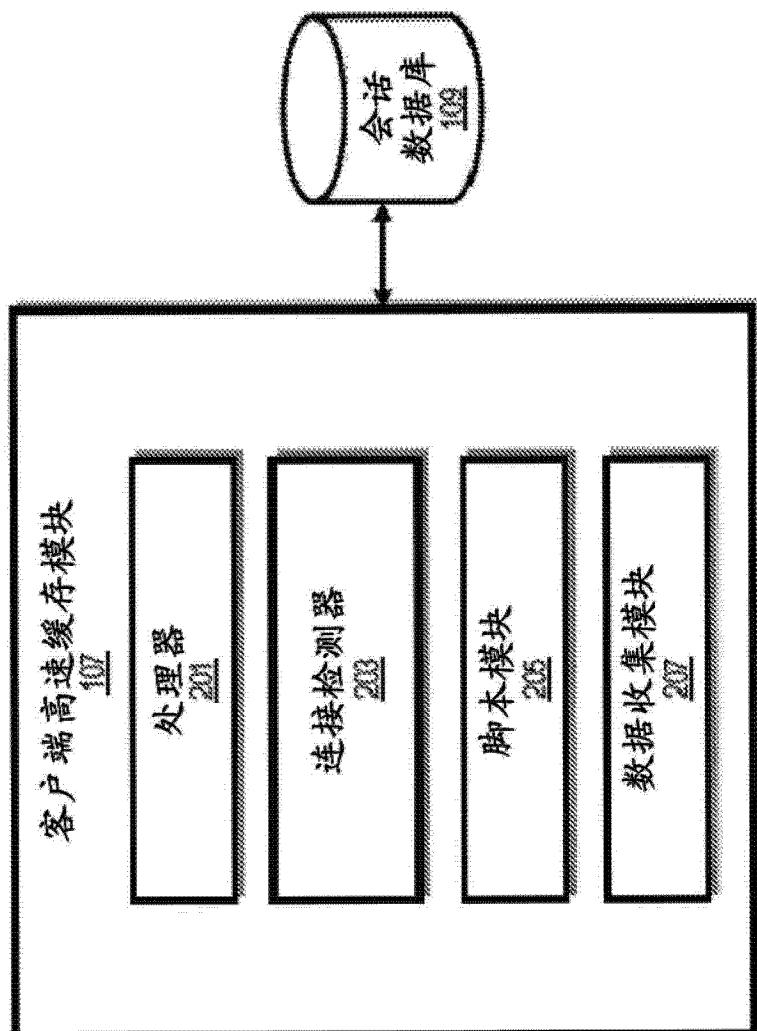


图 2

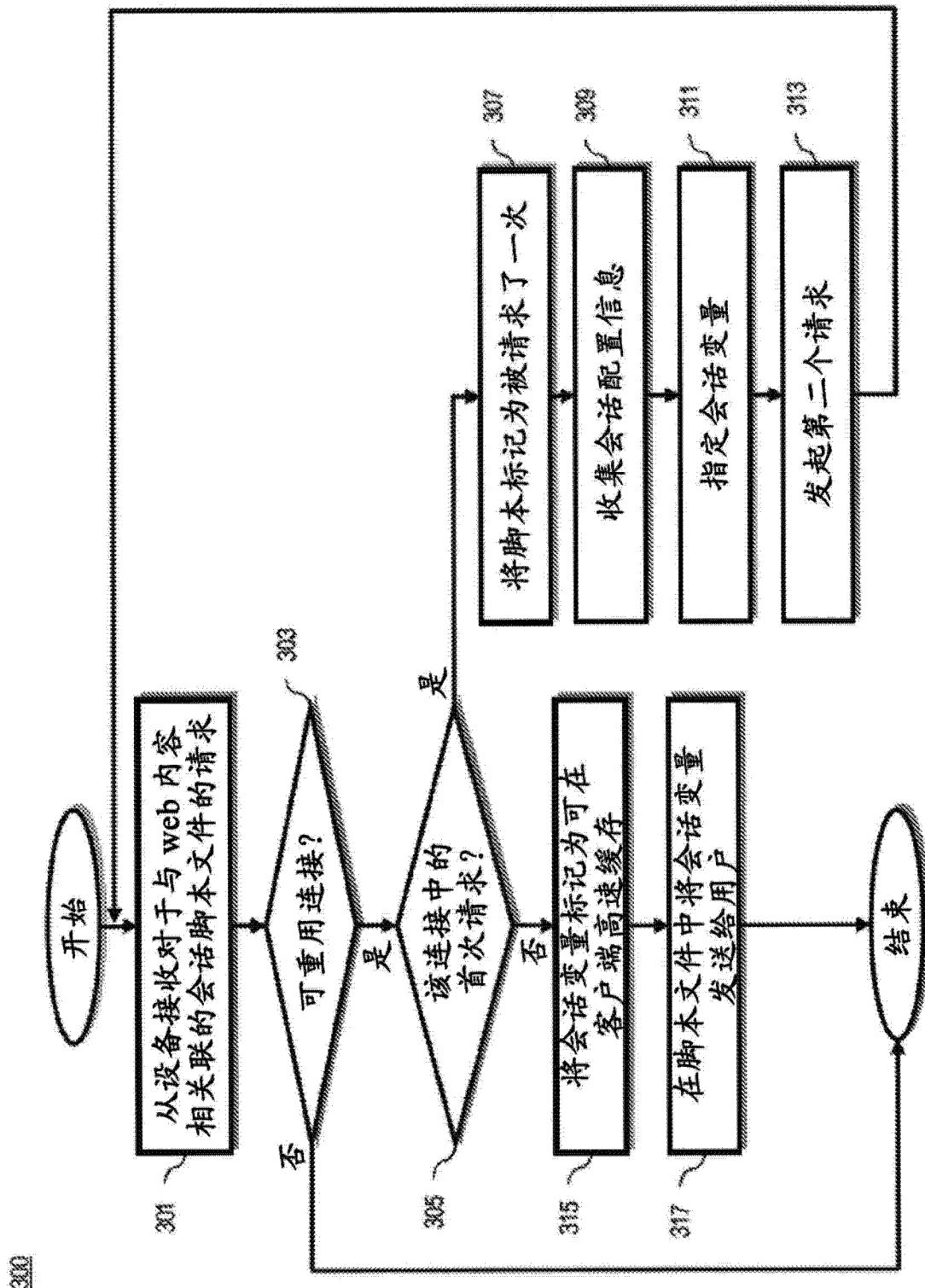


图 3

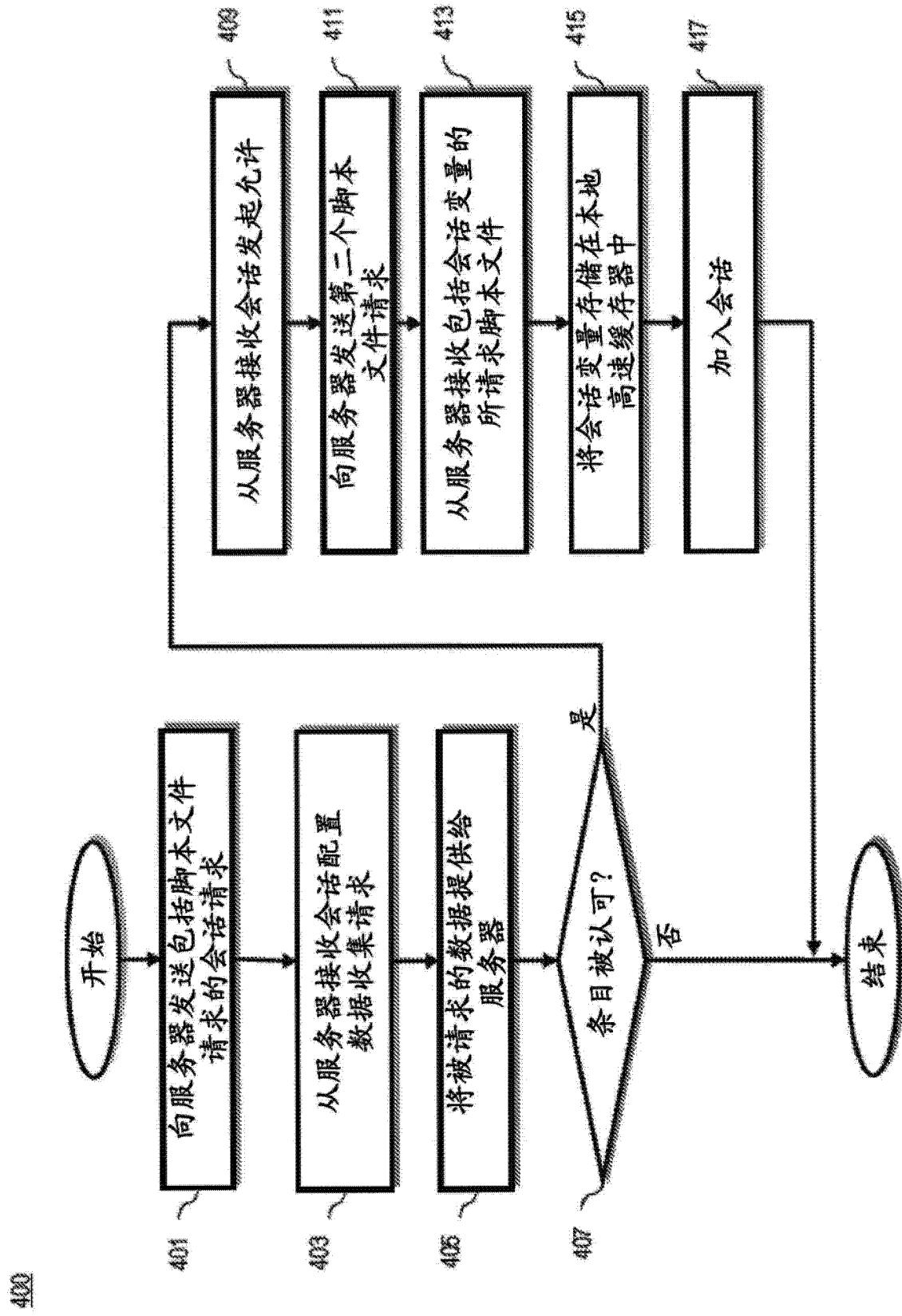
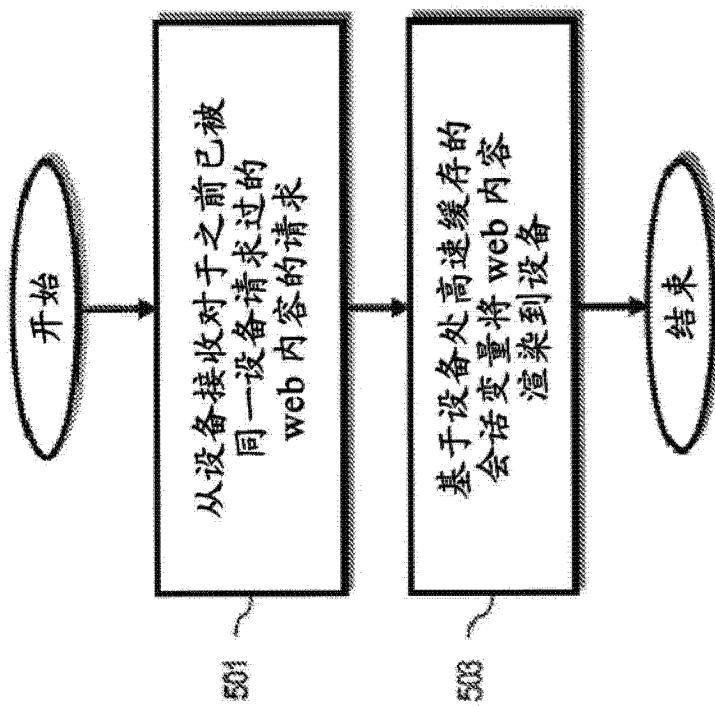


图 4



501

501

503

图 5

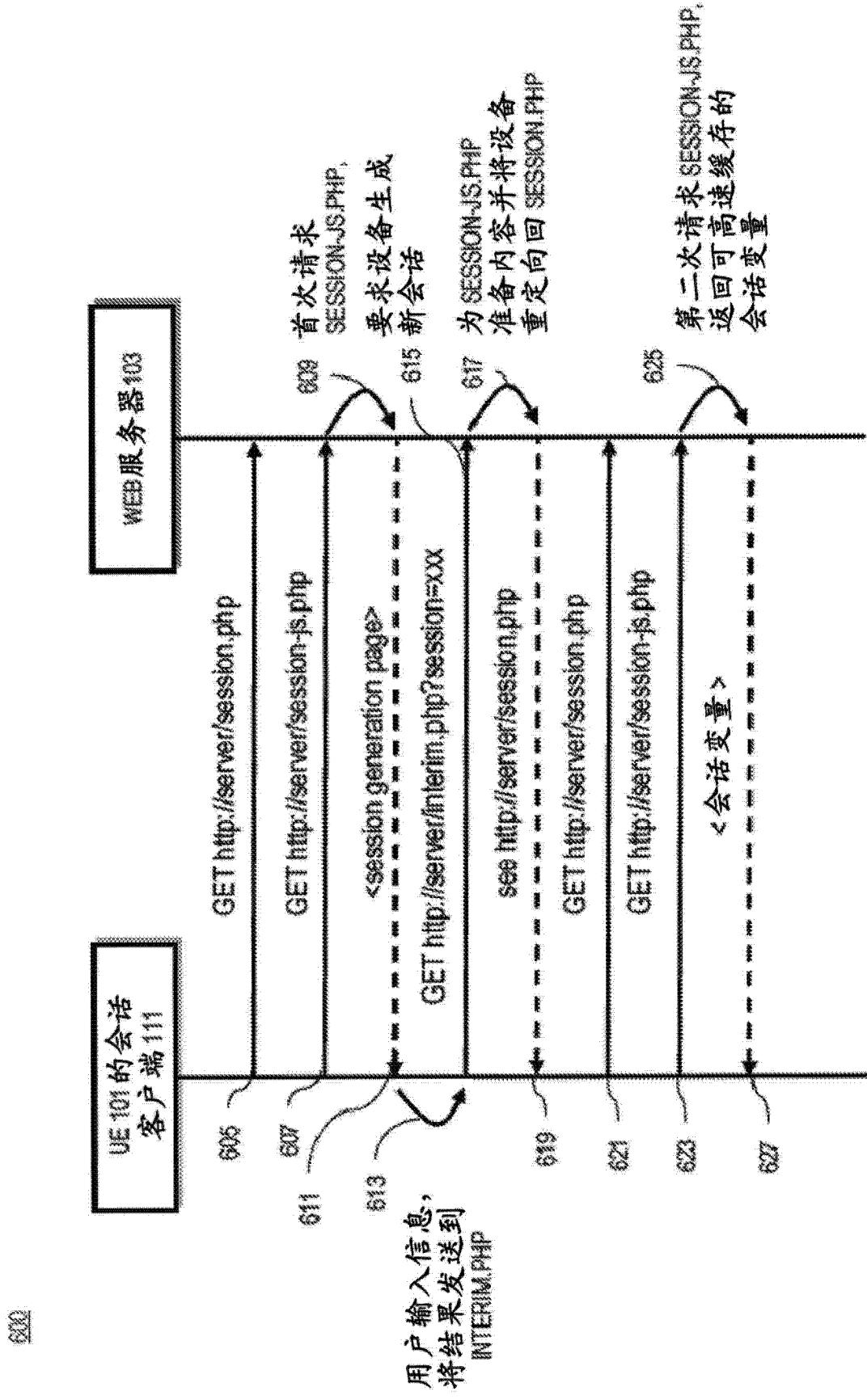


图 6

用户 ID:	<input type="text"/>
密码:	<input type="text"/>
邮政编码:	<input type="text"/> 100000
年龄:	<input type="text"/> 18
爱好:	<input type="text"/> 篮球
	<input type="button" value="提交"/> <input type="button" value="取消"/>

100000

图 7

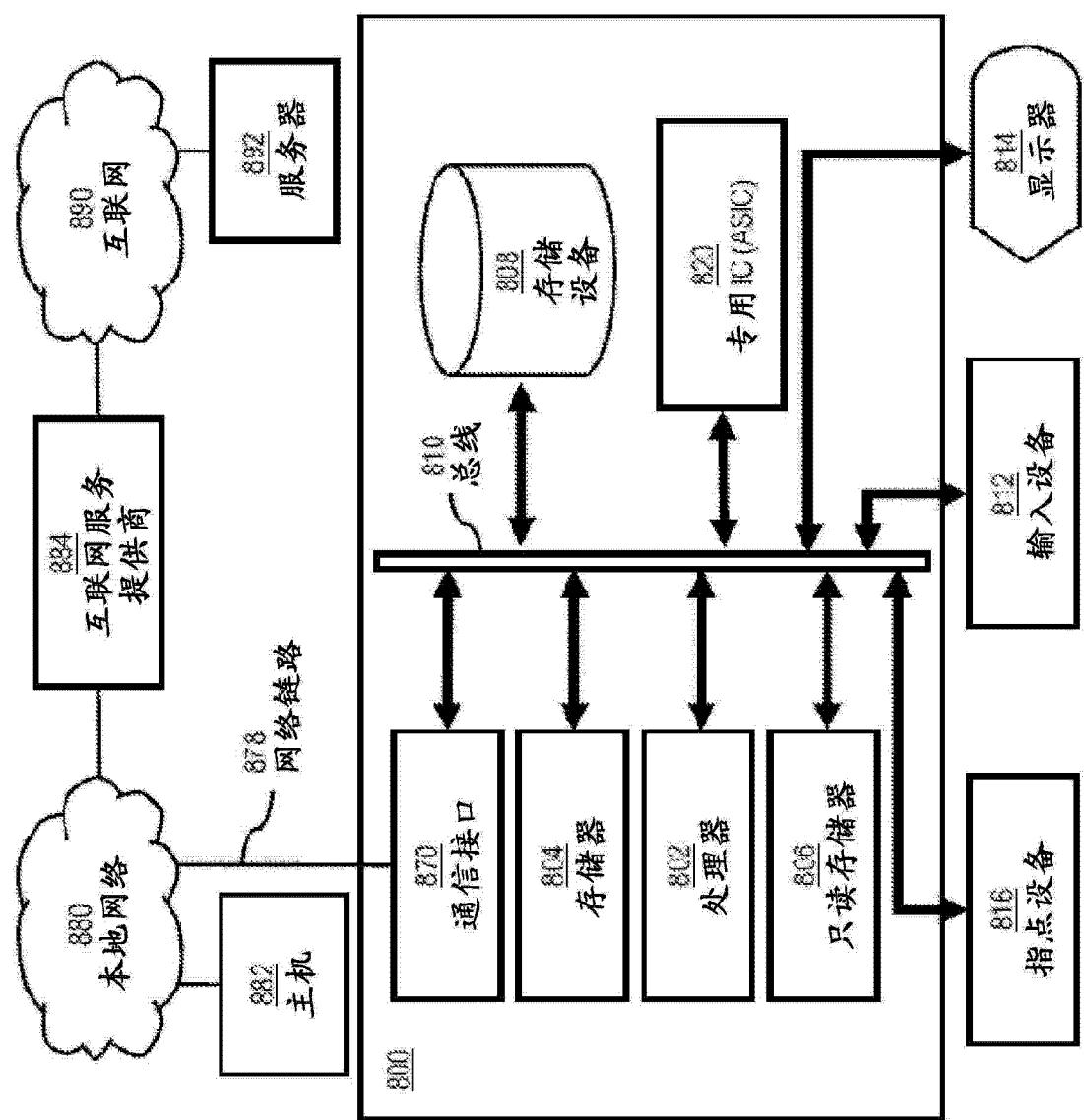


图 8

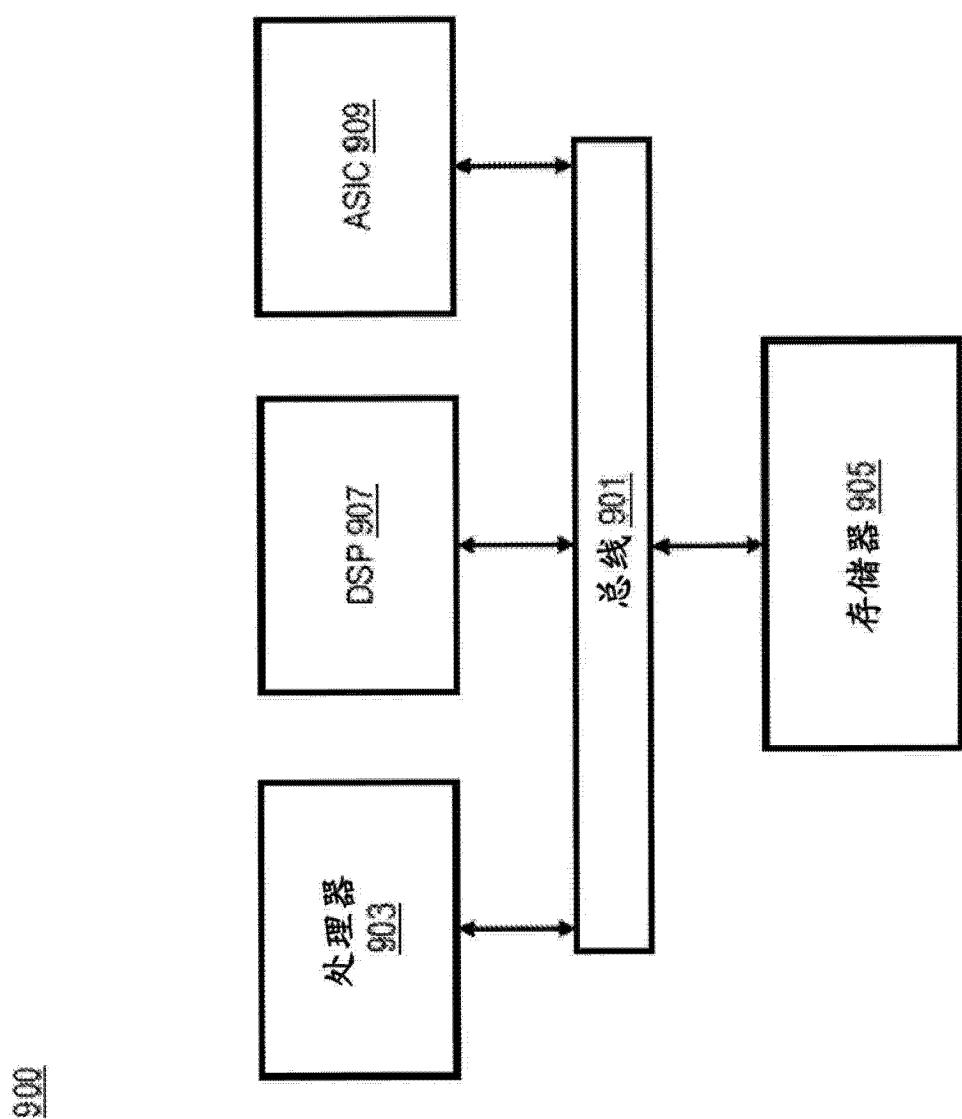


图 9

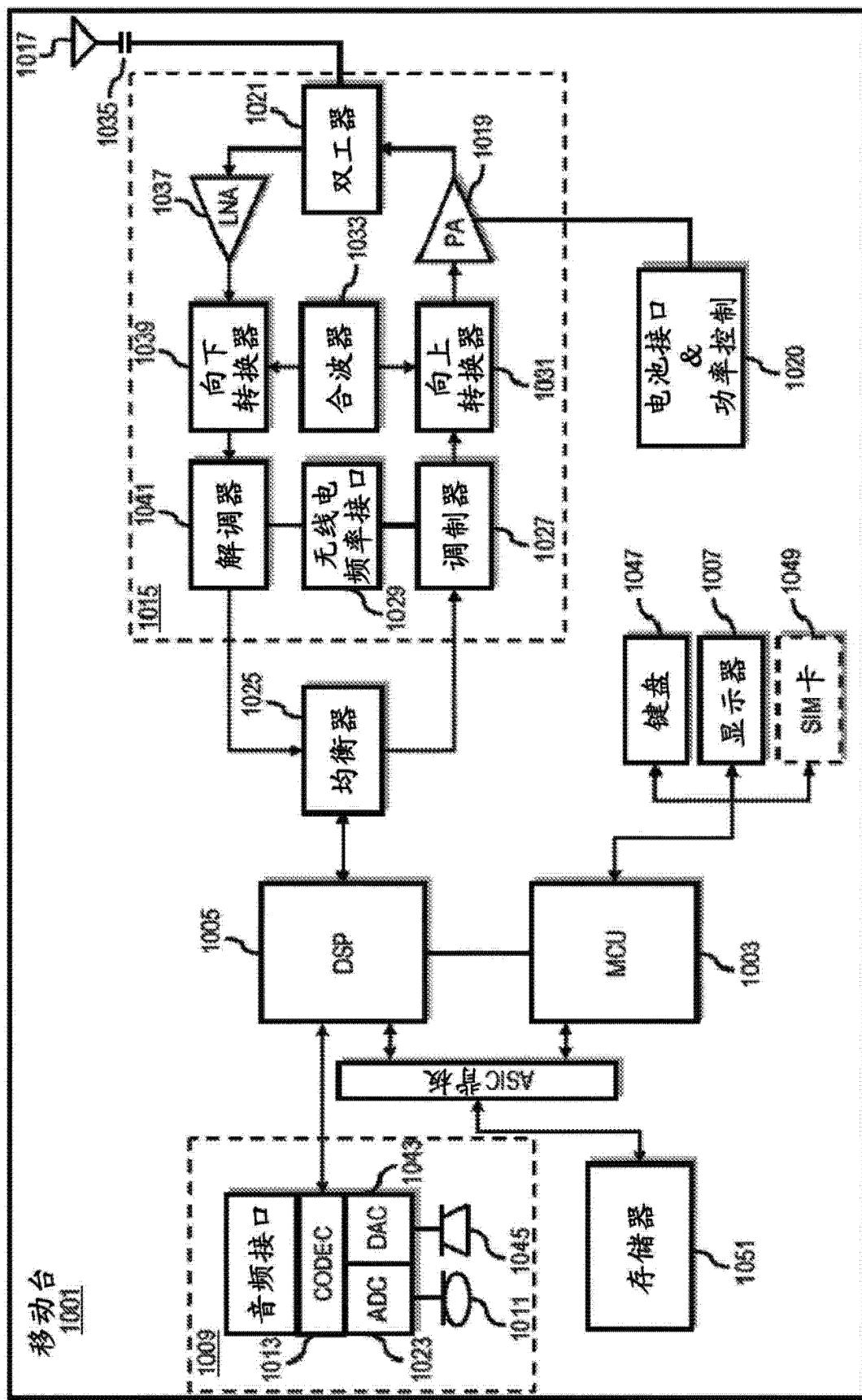


图 10