



(12) 发明专利

(10) 授权公告号 CN 1649324 B

(45) 授权公告日 2010.05.05

(21) 申请号 200510004379.4

(22) 申请日 2005.01.20

(30) 优先权数据

10/764,754 2004.01.26 US

(73) 专利权人 朗迅科技公司

地址 美国新泽西州

(72) 发明人 维姆·海伦塔尔

弗兰斯·约塞夫斯·潘肯

(74) 专利代理机构 北京市中咨律师事务所

11247

代理人 杨晓光 刘薇

(51) Int. Cl.

H04L 12/54 (2006.01)

(56) 对比文件

US 2002/0101879 A1, 2002.08.01, 全文.

臧磊. Parlay API 的构成. 邮电商情 2002 年第 12 期. 2002, (2002 年第 12 期), 39-43.

柯国富, 马玉祥. 基于 Parlay API 规范的下一代网络业务架构. 电子科技 2002 年第 1

期. 2002, (2002 年第 1 期), 22-24.

Andrew J. Bennett, Musa R. Unmehopa, Kumar V. Vemuri. The Parlay Proxy Manager—Architecture Considerations. Bell Labs Technical Journal 4. 2003, 7(4), 91-104.

审查员 赖异

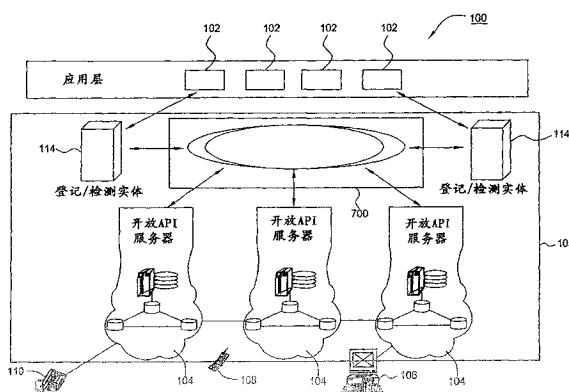
权利要求书 2 页 说明书 6 页 附图 7 页

(54) 发明名称

操作带有代理的开放 API 网络的方法和装置

(57) 摘要

一种电信系统、方法、设备、和计算机可读介质,其实现一开放应用程序接口 (API) 系统,其中开放 API 服务器通过一个新的代理向应用递送并接收 API 命令。该代理辅助实施各种管理功能,诸如为应用分配开放 API 服务器以及监视整个系统,和服务合同控制,其都以透明的方式进行。该代理使得电信系统能够处理地理上分散的应用和开放 API 服务器。所有的 API 命令都能够,但是不必,通过该代理。



1. 一种基于开放 API 的系统,包括:

开放 API 服务器,用于与用户设备通信并发送和接收应用程序接口命令;以及

代理,用于从所述开放 API 服务器接收应用程序接口命令,用于向一应用发送从所述开放 API 服务器接收的应用程序接口命令,用于从所述应用接收响应,以及用于向所述开放 API 服务器发送从所述应用接收的所述响应;

其中所述代理处理服务合同项目以确定多个开放 API 服务器的每一个的本地服务合同实施参数以及向各个开放 API 服务器发送各自的本地服务合同实施参数,其中发送到每个开放 API 服务器的所述本地服务合同实施参数指导该开放 API 服务器实施本地服务合同项目;并且

其中每个开放 API 服务器基于各自的本地服务合同实施参数控制发送应用程序接口命令。

2. 根据权利要求 1 所述的系统,其中所述本地服务合同实施参数与服务合同有关。

3. 根据权利要求 1 所述的系统,还包括登记和检测设备,其接收服务合同项目作为控制参数。

4. 根据权利要求 1 所述的系统,还包括计算机可读介质,用于存储至少部分控制所述代理产生所述本地服务合同实施参数的程序信息。

5. 根据权利要求 1 所述的系统,其中,每个开放 API 服务器用于基于服务的使用,请求修改本地服务合同实施参数。

6. 一种基于开放 API 的系统,包括:

电信网络;

开放 API 服务器,用于在所述电信网络上发送并接收应用程序接口命令;以及

代理,用于从所述开放 API 服务器接收应用程序接口命令,并且将接收到的应用程序接口命令从所述开放 API 服务器发送到应用;

其中,所述代理处理服务合同项目以确定多个开放 API 服务器的每一个的本地服务合同实施参数以及向各个开放 API 服务器发送各自的本地服务合同实施参数,其中发送到每个开放 API 服务器的所述本地服务合同实施参数指导该开放 API 服务器实施本地服务合同项目;并且

其中每个开放 API 服务器基于各自的本地服务合同实施参数控制发送应用程序接口命令。

7. 根据权利要求 6 所述的系统,其中所述代理还用于根据系统的变化改变其选择,监视所述系统的状态,以及从与所述应用进行通信动态地改变到与另一应用进行通信。

8. 根据权利要求 6 所述的系统,其中所述开放 API 服务器绕过所述代理并且向一应用直接发送应用程序接口命令以防止代理成为通信瓶颈。

9. 根据权利要求 8 所述的系统,其中所述代理阻止预定应用程序接口命令传播。

10. 一种操作电信网络的方法,包括:

获得服务合同项目;

处理服务合同项目以确定多个开放 API 服务器的每一个的本地服务合同实施参数;

向各个开放 API 服务器发送各自的本地服务合同实施参数,其中发送给每个开放 API 服务器的本地服务合同实施参数指导该开放 API 服务器实施本地服务合同项目;

从所述多个开放 API 服务器接收应用程序接口命令 ; 以及
将所接收的应用程序接口命令递送给至少一个应用 ;
其中, 每个开放 API 服务器基于各自的本地服务合同实施参数, 发送应用程序接口命令。

操作带有代理的开放 API 网络的方法和装置

技术领域

[0001] 本发明的实施例涉及电信网络。更具体来说,本发明的实施例涉及支持开放 API 资源的电信网络。

背景技术

[0002] 电信网络的运营商通常面临不得已向提供基于通信的应用的新公司开放其网络。这种勉强之所以或多或少合理的存在是由于网络运营商会冒着拆分其自身应用的风险,特别是,如果价格成为其自身应用和新的公司所提供的应用之间的主要差别时。实际上,如果电信网络运营商不能与新公司所提供的具有低开销的应用相媲美的话,其将冒着仅仅成为“比特传送器”的风险。

[0003] 此外,如果电信网络运营商向新的应用和公司开放其网络,由于必须支持由例如服务提供者或软件供应商等第三方著述并维护的应用所驱动的网络需求,网络运营商将面临失去对其网络的控制的局面。

[0004] 但勉强的开放其网络可能是合理的,如果电信网络运营商不这样做,其就会承担失去有利市场的后果,而这一市场则可能由新公司和应用占领。由于大多数新公司都在传统的电信网络边界之外操作,新的应用就可能创建通常不需要网络的完全新的服务来争得客户。此外,政府命令的力量——以期望提高电信市场的竞争——也迫使网络运营商开放其网络。

[0005] 在一个技术方案中,网络和应用之间的事务处理得益于一组形成应用程序接口(API)的接口和数据类型。如果这些 API 被标准化并且很宽泛的适于应用,这些 API 就被称为开放 API。开放 API 通常由精细定义 API 并颁布其应用的软件组织来确定。现在已有多组开放 API,例如 Parlay/OSA API,其最初在 Parlay 组中定义并且在 3GPP 和 ETSI 文件中被标准化。各 Parlay/OSA API(其中 OSA 表示开放服务结构,Open Service Architecture)形成了一组 9 个正交的服务性能功能(SCF),每一个表示一个不同的电信领域:呼叫控制、用户交互、移动性、终端性能、数据话路控制、类属消息、连接管理、存在/可用性以及收费。

[0006] 这样,现有技术中基于 API 的系统包括一组开放 API、一组经由服务器发送并接收信息的用户、传输所述信息的电信网络、以及通过开放 API 接收信息的第三方拥有的应用,基于这些 API 运行应用,并且通过开放 API 将应答信息发送回网络。

[0007] 基于现有技术 API 的系统虽然很有用,但是确有很大的局限性。首先,在现有技术基于 API 的系统中,开放 API 服务器和应用之间为一对一的关系。为了克服这一限制,基于现有技术 API 的系统已经包括一个登记和检测实体,其由开放 API 服务器使用来通报其自身,并且由应用使用来检测 API 是否可用。当有多个登记和检测实体时,产生的一个最基本的问题是开放 API 服务器应该使用哪个登记和检测实体。为了避免这一问题,开放 API 服务器可以向所有可用的登记和检测实体通报其自身。但是,这样就向多个登记和检测实体分配了一个开放 API,这同样会产生一个问题:必须知道整个系统的配置,但这非常难以确定。

[0008] 另一个现有技术中基于 API 的系统所产生的问题是：当一个应用检测到开放 API 为实现其服务将使用什么时，这一关系就成为静态的。尽管起始的开放 API 服务器可能在一个特定时间是有用的，但随后发生的事情，诸如开放 API 服务器的失败，将会产生巨大的问题。在这种情况下，提供了特定服务的应用将不再允许在没有由该应用初始化的恢复通话的情况下使用。此外，另一个在现有技术中基于开放 API 的系统的问题是很难实施服务层协议。

[0009] 从上述（以及其他）限制来看，希望提出一个新的基于开放 API 的系统。这一系统包括开放的应用和开放的 API 服务器，但电信网络在开放应用和开放 API 服务器之间插入了代理设备。该代理能够处理启动来避免重叠，确定哪个开放 API 服务器（或者服务器组）将执行特殊 API 事件或方法调用，平衡各种开放应用和开放 API 服务器之间的通信负载，并且在开放应用和开放 API 之间分配各事件。

[0010] 尽管代理是对基于开放 API 的网络比较有效的添加，但是所提出的代理不能够应对在成功的系统中的其他可预见的问题。由此，能够解决在基于开放 API 的系统中其他问题的代理将是十分有益的。另外这种有益效果将可能是一个带有解决基于开放 API 系统中其他问题的代理的电信网络。此外，解决基于开放 API 系统中其他问题的代理的操作方法也是很有益的。一种存储有操作解决基于开放 API 的系统中其他问题的代理的计算机程序的计算机可读介质也是十分有益的。

发明内容

[0011] 目前与现有技术相关的缺陷可以通过克服基于开放 API 系统的限制的方法来由操作通信网络的新的方法来解决。

[0012] 目前与现有技术相关的缺陷可以通过新的代理以及带有所述代理的通信网络来解决，以克服基于开放 API 系统的限制。

[0013] 目前与现有技术相关的缺陷可以通过存储有操作所述代理的计算机程序的计算机可读存储介质以及由此的通信网络来解决，以克服基于开放 API 系统的限制。

附图说明

[0014] 本发明的原理可以通过参考随后的细节说明结合附图容易的理解，其中：

[0015] 图 1 示出了根据本发明原理的基于 API 的系统概貌；

[0016] 图 2 示出了代理的总体视图；

[0017] 图 3A 示出了实施服务合同控制的系统；

[0018] 图 3B 示出了实施服务合同控制的流程图；

[0019] 图 4A 示出了地理上分配的基于开放 API 的系统；

[0020] 图 4B 示出了登记开放 API 实体；

[0021] 图 4C 示出了监视开放 API 系统的代理；以及

[0022] 图 5 示出了根据本发明原理的可选基于 API 的系统。

[0023] 为了便于理解，在可能的情况下，使用相同的参考标记来标识各图中相同的元素。

具体实施方式

[0024] 本发明针对一种在应用和开放 API 服务器之间包含代理的新的基于开放 API 的方法、装置、计算机可读介质、以及系统。根据本发明的开放 API 系统可以处理两种类型的信息流，在网络中生成并且流入到应用的信息流、以及从应用流出到网络的信息流。虽然两种流都通过 API 来进行通信，但其中第一种类型被称为事件，而第二种类型被视为方法调用。

[0025] 图 1 示出了根据本发明的原理的基于开放 API 的系统 100。该系统 100 包括应用 102 以及开放 API 服务器 104，其通过开放应用程序接口传递信息。应用 102 可以是程序、企业应用类型系统、或使用 API 进行操作访问网络资源的其他资源。开放 API 服务器 104 是连接到任何宽范围的用户设备的通信节点，所述用户设备诸如计算机 106、手持设备 108、或者电话系统 110。相应的，用户设备可被理解是任何渴望运行应用 102 的设备。当用户设备中的一个连接到开放 API 服务器 104 而后其以下述方式向应用 102 传递信息时，执行由应用 102 提供的服务。

[0026] 如图所示，在开放 API 服务器 104 和应用 102 之间是一个代理 700。开放 API 服务器 104 和代理 700 可以是公用自身网络 103 的一部分。此外，虽然应用 102 在应用层上示出，但一个或更多的应用 102 也有可能由网络 103 的属主拥有或操作。正如随后将要具体描述的那样，代理 700 透明的处理开放 API 服务器 104 和应用 102 之间的通信。为了这一目的，应用 102 初始验证并随后向代理 700 登记其存在，这由连接登记和检测设备 114 的应用 102 来实现，所述登记和检测设备 114 接收 - 或者能够从一验证机制确定 - 代理 700 识别、接受、以及使用应用 102 所需的登记信息（诸如 IP 地址、启动条件、地理位置、容许的 API，等）。

[0027] 应用 102 的登记主要发生在当应用 102 通报其启动条件时。在应用 102 被验证后，它就能够发现、选择、以及开始使用由开放 API 服务器 104 提供的服务。通过登记，代理 700 将确定所有开放 API 服务器 104 的性能并且随后将这些性能结合起来形成性能超集。代理 700 随后向登记和检测设备 114 登记所述性能超集，使得登记和检测设备 114 有可能向被验证的正在检测服务的应用 102 公告更一般的一组服务。这同样提供了开放 API 服务器 104 仅仅在单个位置登记的能力，并且同样提供了监视并且保持开放 API 网络的中央地点，其将在随后描述。

[0028] 图 2 示出了代理 700 的实施例的一个更高层的方框图。代理 700 包括处理器 710 以及计算机存储器 720 用于存储控制程序 725 和数据结构 727。处理器 710 与诸如电源、时钟电路、缓冲存储器等的传统配套（support）电路 730，以及帮助执行存储在存储器 720 中的软件例程的电路共同运作。这样，考虑在此讨论的一些处理步骤作为软件过程可以由硬件实施，例如，作为与处理器 710 共同运作来执行各种步骤的电路。代理 700 也包括输入 - 输出电路 740，其形成一个对整个电信网络、应用 102 和开放 API 服务器 104 的接口。

[0029] 虽然根据本发明代理 700 被视为一个被编程执行各种控制功能的通用计算机，本发明也可由硬件实施，例如，作为一个专用集成电路（ASIC）。这样，此处描述的处理步骤有望宽泛的理解为相等的由软件、硬件、或者其结合来实现。此外，如果由软件或固件实施，则代理 700 包括存储有可由处理器 710 运行和 / 或访问的信息的计算机可读介质 750。不论由硬件还是软件，或是硬件 / 软件的结合来实施，代理 700 都操作来实现下述功能。

[0030] 由代理 700 实现的第一个功能在于实现两个或多个开放 API 服务器 104，用户、应用 102，以及电信网络运营商之间的服务合同。可能作为服务合同的一部分的参数实例包

括由开放 API 服务器 104 许可的资源的使用,举例来说,呼叫的最大次数,每次呼叫所允许的最大呼叫支路数,开放的最大信箱数,帐单,时间限制,以及所允许的 API。在一些实施例中,代理 700 动态的相开放 API 服务器 104 分布服务合同。

[0031] 图 3A 示出了用于通过代理 700 实施服务合同控制的系统 200,同时图 3B 示出了操作系统 200 的流程图。下文将参考这两张图来描述。系统 200 包括登记和检测设备 114,其操作性地连接到一个企业操作者 202 并且通过数据库 204 连接到代理 700。然而,应该理解,数据库 204 是可选择的:一些实施例将不包含数据库 204。现在参考图 3B,服务合同控制的方法 245 从步骤 247 开始并且流入步骤 248 以获得并登记服务合同控制信息。为了实现这一目的,企业操作者 202 提供了服务合同的项目作为登记和检测设备 114 的一组控制参数,在步骤 250 将控制参数存储到数据库 204 中。接下来,在步骤 251,代理 700 取回所述控制参数,并且在步骤 252 计算如何将全局合同信息作为本地合同分配给相关开放 API 服务器 104(由服务合同项目所影响)。而后,在步骤 253,通过代理 700 在步骤 254 发送各开放 API 服务器 104 的实施参数,本地合同被发送到开放 API 服务器 104。

[0032] 在开放 API 服务器 104 得到其实施参数后,在步骤 255 一个触发器可导致本地开放 API 服务器 104 确定其本地服务合同是不足的。例如,如果本地开放 API 服务器 104 被限制为 10 个呼叫,当其处理了 8 个呼叫时,开放 API 服务器 104 将设置一个触发器来指示其需要授权来处理多于 10 个的呼叫。在这种情况下,在步骤 256,本地开放 API 服务器 104 向代理 700 发送一个请求要求修改其服务合同。同时,在步骤 257,代理 700 一直监视开放 API 服务器 104 来识别这一请求。在步骤 258,当已经接收到修改的请求后,代理 700 在步骤 258 查询相关开放 API 服务器的当前本地使用。一旦确定了所述使用,循环返回步骤 252 来重新计算相关开放 API 服务器的本地合同。这样,代理 700 就可以在保持服务合同的情况下根据当前使用分配资源。

[0033] 应该理解,每个实体都可能对服务合同项目作出不同的反应。例如,开放 API 服务器 104c 的服务合同可能指示没有时间关键性的信息将传递至或来自开放 API 服务器 104c。相应的,代理 700 可以给其他开放 API 服务器 104a 和 104b 的 API 一个更高的优先级。

[0034] 当已向实体提供了实施参数后,在步骤 255 各种实体将实施其服务项目的那部分。在方法 256 中,如果开放 API 服务器 104 确定是特殊的一组应用的本地服务合同请求了一个更新,这一开放 API 服务器 104 就可以询问代理,请求更新合同。如果一组应用的服务合同停止了,所述方法则在步骤 257 处停止。例如,当用户连接一个开放 API 服务器 104 时,该服务器搜索其服务合同参数并且确定它是否能够构处理由用户 206 请求的特殊的呼叫或者服务。例如,如果开放 API 服务器 104a 有最大呼叫次数,所有超过这一次数的呼叫都将被拒绝或者重定向到另一个开放 API 服务器 104。

[0035] 随着开放 API 模型变得越来越普遍,开放 API 实体的数目将增加。这将引发模型的管理和整合问题。这是一个特殊的挑战,由于一个开放 API 系统可以由存在于非常不同的地理位置的各种实体组成。因为必须监视开放 API 实体的性能来保障全系统的顺应性,并且,如果需要,也必须采取适当的校正操作,所以不同的地理位置将产生问题。校正操作时一个主要的考虑,因为在大的网络中全系统的改变将是非常困难的、耗费时间的、并且昂贵的。代理 700 对这一问题提供了一个有效的解决方案。此外,代理 700 还是以透明的方式实现的。

[0036] 图 4A 示出了欧洲的地理地图其中各种开放 API 服务器 104 在整个大陆上分布。所有这些开放 API 服务器都由同一个代理 700 来管理。应该理解,代理 700 不必须是一个单独的实体,而是可以由合作形成代理 700 的相互连接的设备、系统和网络形成的一个网络构成。集中管理该代理 700 极大的简化了全系统的改变,因为这些改变都可以由代理 700 来发布。

[0037] 图 4B 示出了代理 700 是如何成为中央实体。在步骤 505,代理 700 从开放 API 服务器 104 接受登记和取消登记。在步骤 507,代理 700 确定所有开放 API 服务器 104 的服务性能,并且而后确定这些性能的一个超集。代理 700 随后向所有已知的登记和检测设备 114 登记这一超集。通过充当一个中央登记点,代理 700 获得了所有开放 API 服务器 104、登记和检测实体 114、以及在开放 API 网络中的应用的状态。这就使得代理 700 充当了整个系统的管理器和整合器。

[0038] 图 4C 示出了代理 700 如何管理和整合系统的方法 301。如图所示,方法 301 在步骤 303 处开始并且流入步骤 315 来监视整个开放 API 系统的操作。在步骤 317,如果确定开放 API 系统 200 将正常工作,方法返回步骤 315,由此继续监视整个系统的操作。然而,如果在步骤 317 检测到异常,代理 700 将流入步骤 319,在此进行适当的校正操作。适当的操作可能包括重新启动登记和检测设备 114,将所检测到的网络中的问题通知给应用 102,以及/或当其被再次激活时,向开放 API 服务器 104 重新分配启动条件。注意,如果代理找到了替代的开放 API 服务器 104 来保证开放 API 系统 200 的完整性,则可以延迟或不执行所述向应用 102 通报异常。

[0039] 代理 700 可以采取的一个重要的校正操作是充当防火墙来阻止不适当的或危险的 API 在开放 API 系统中传播。这将是特别有效的,如果登记和检测设备 114 在一个渴望为第三方开放其网络资源的外国网络运营商的领域内操作时,所述第三方可能对 API 进行改进而导致本地或整个系统故障。有这样的代理 700 充当过滤器来阻止所述 API 将是特别有益的。

[0040] 如上所述,代理 700 将感知到其他开放 API 系统实体并且能够直接为特定应用 102 分配特定的开放 API 服务器 104,不仅是初始的,而且是动态的。分配也可以被改变以反映网络条件,诸如应用 102 和开放 API 服务器 104 的添加,或者如系统故障结果。这一动态分配的性能显著的提高了网络的可用性。

[0041] 虽然上述内容使人认为在应用 102 和开放 API 服务器 104 之间的所有通信都将通过代理 700,但这并不是必须的。图 5 示出了一个开放 API 系统 400,其中特定 API 呼叫绕过代理 700 并且直接在开放 API 服务器 104b 和应用 102b 之间运行。如图所示,特定呼叫,指定的 Ip 呼叫 (IpCall),直接在线路 401 上通信。然而,其他 API 呼叫则由代理 700 处理。直接路由可以防止代理 700 成为其性能的瓶颈。

[0042] 图 5 也示出了由代理 700 执行的软件 (或硬件) 过程。包括服务合同软件 402,其处理上述的服务合同,负载平衡 (load-balancing) 软件 404,其平衡开放 API 服务器 104 和各种应用 102 之间的通信负载,管理软件 406,其如上所述管理开放 API 实体,以及事件调度程序 408,其控制在开放 API 服务器 104 和应用之间的各种 API 呼叫的调度。出于完整考虑,图 5 还示出了登记和检测设备 114 以及数据库 204。

[0043] 如果代理 700 被恰当的配置,其是透明的:开放 API 服务器 104 不需要知道其正通

过代理 700 进行通信,并且应用 102,也不需要知道其正通过代理 700 进行通信,除非可能在当初始登记时。代理 700 可以提供适当的启动条件,当若干应有都可用时,确定由哪个竞争的应用来接收事件(充当交通警察),将延迟引入在前的事件来避免潜在的问题,提供中央管理功能,实施服务合同,隐藏网络,选择向其他资源中继未决的或是新的请求(由此增加网络的可用性),充当防火墙来保持系统的完整性,从多个资源处收集信息并且在请求时和/或合适的时间分配所述信息,简化全系统的改变和其整合,并且集中监视应用 102 和开放 API 服务器 104。

[0044] 虽然上述内容针对本发明的具体实施例,但在不背离本发明的基本范畴的情况下可以给出本发明其他以及进一步的实施例,所述范畴由所附权利要求限定。

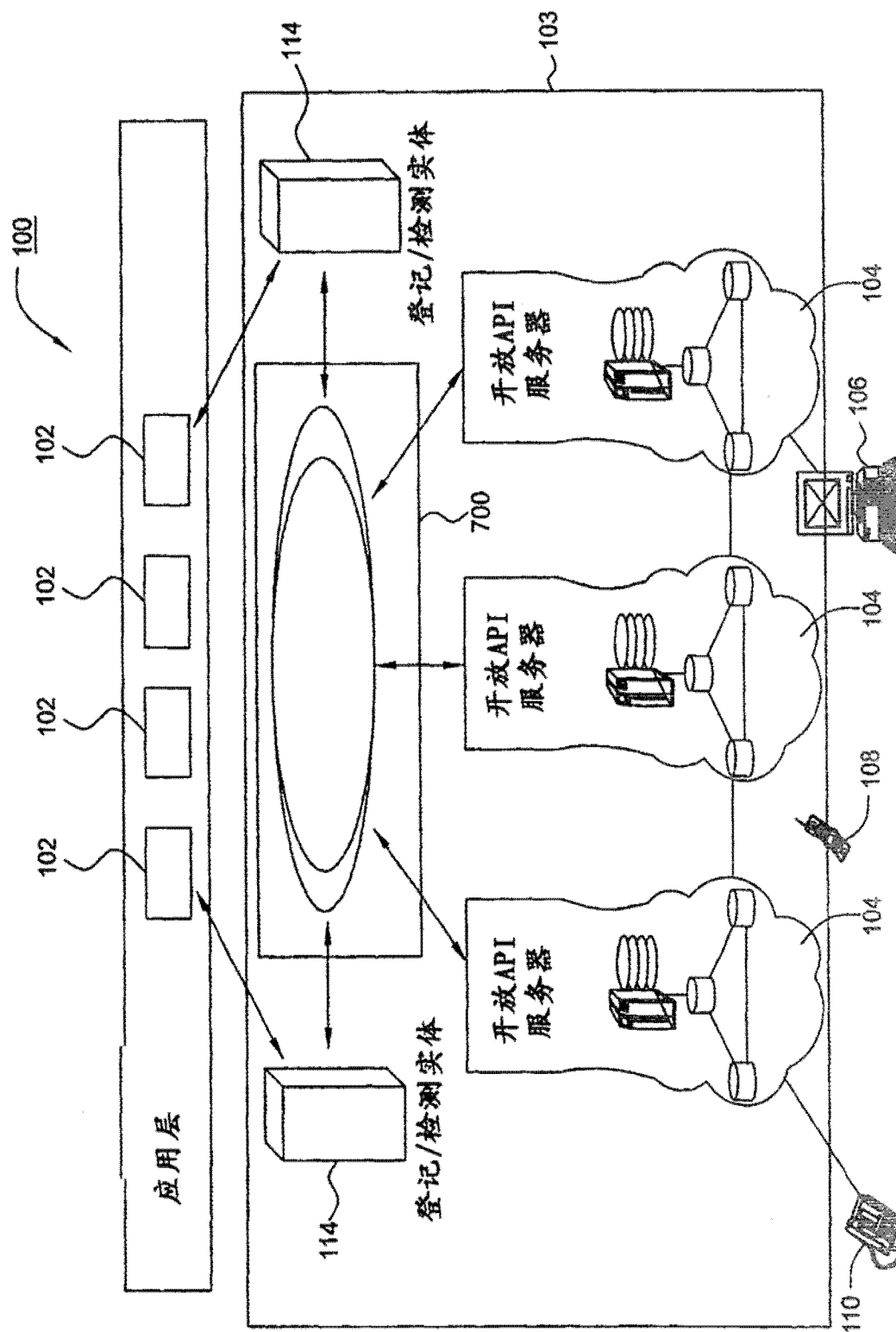


图 1

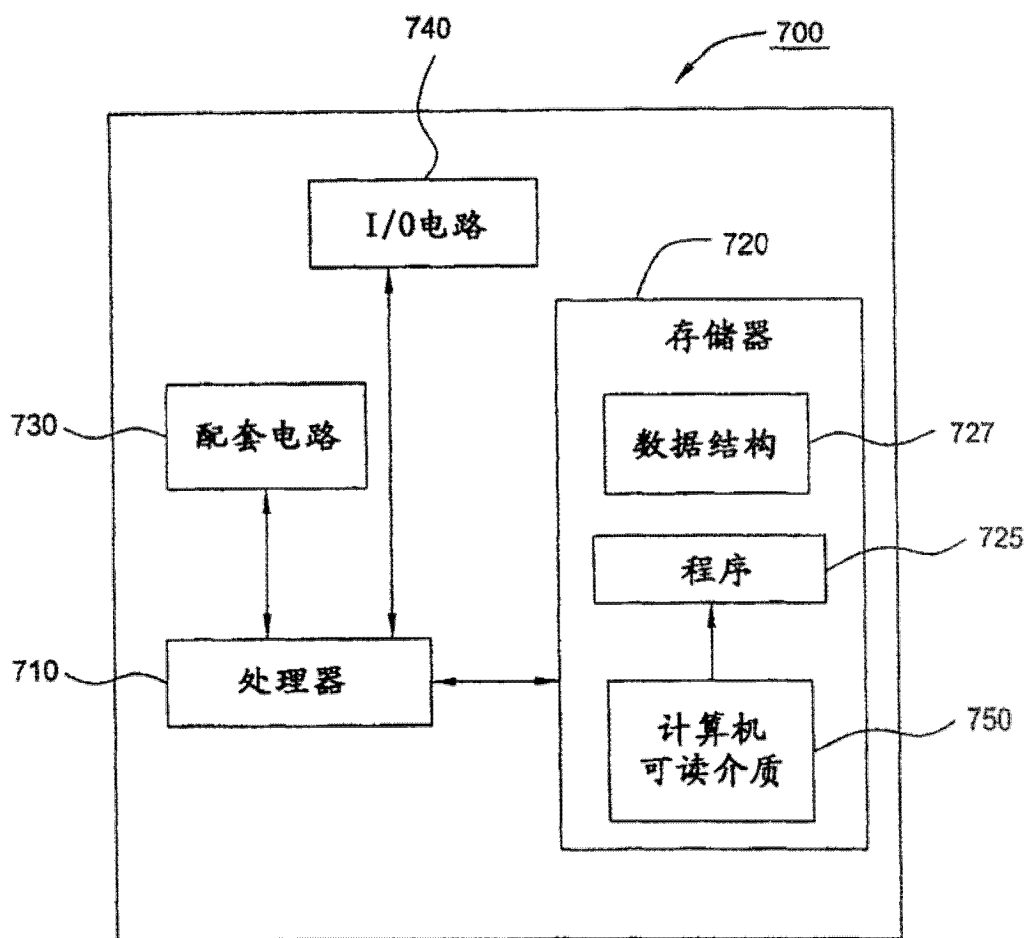


图 2

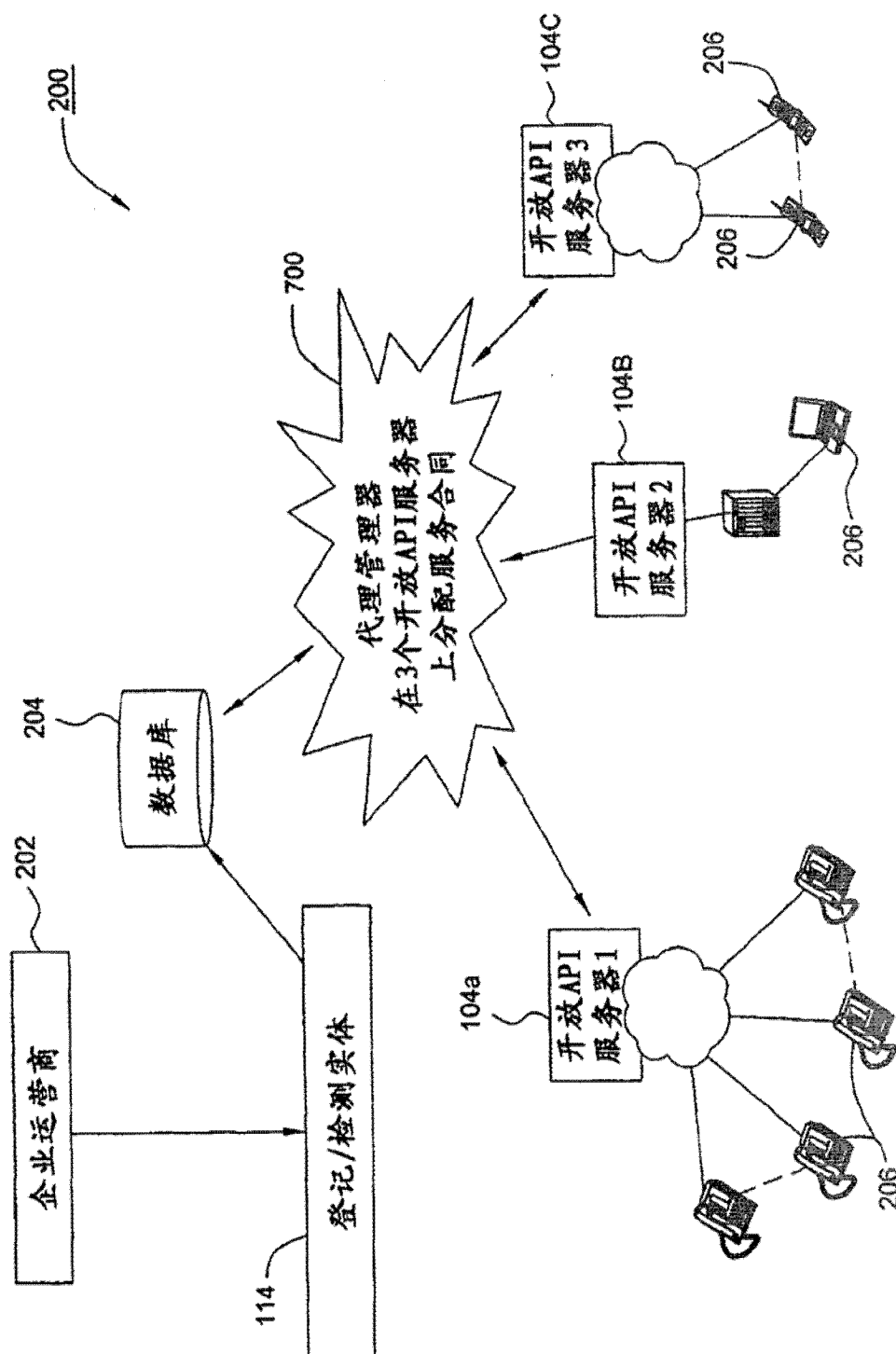


图 3A

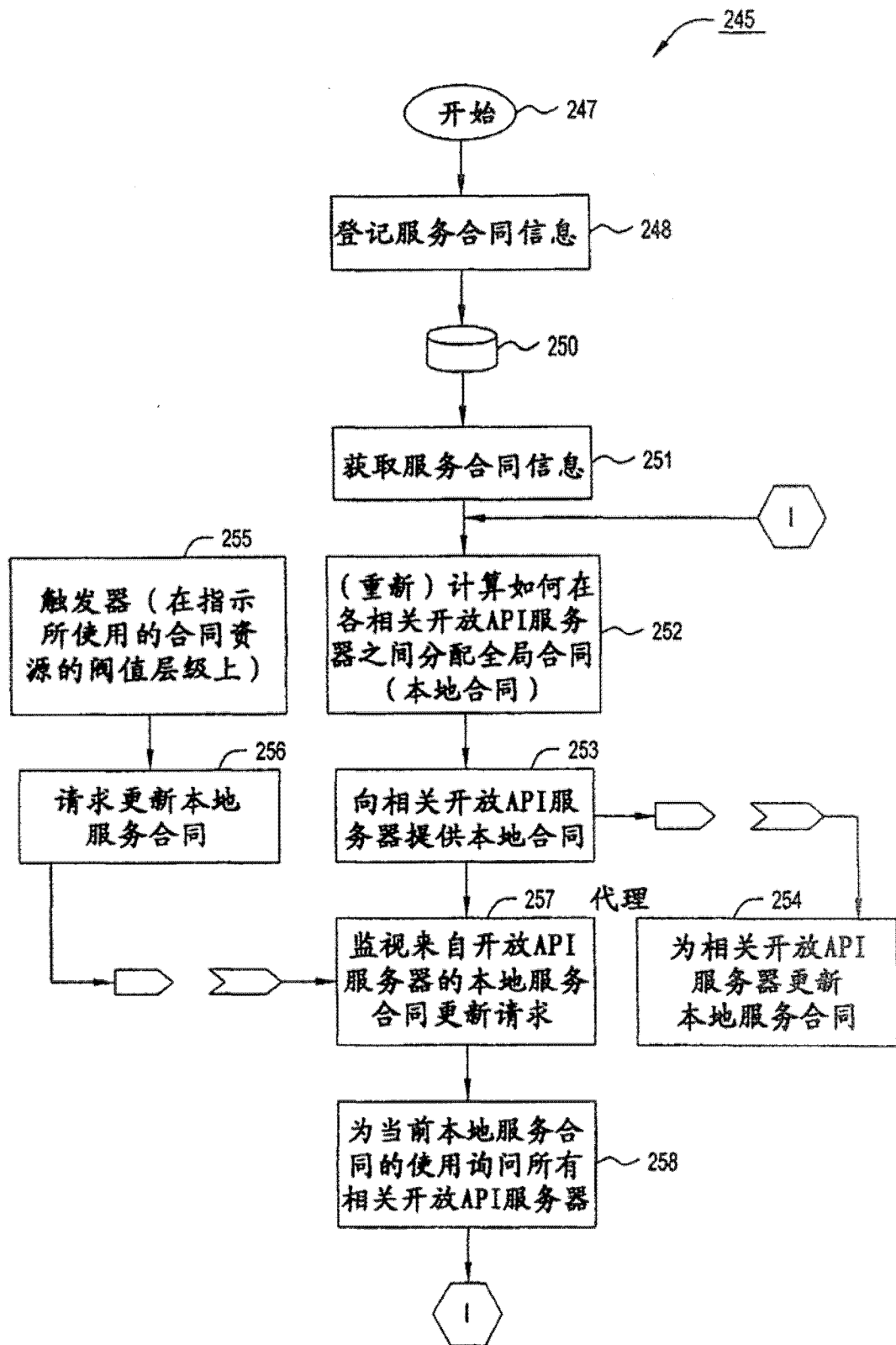


图 3B

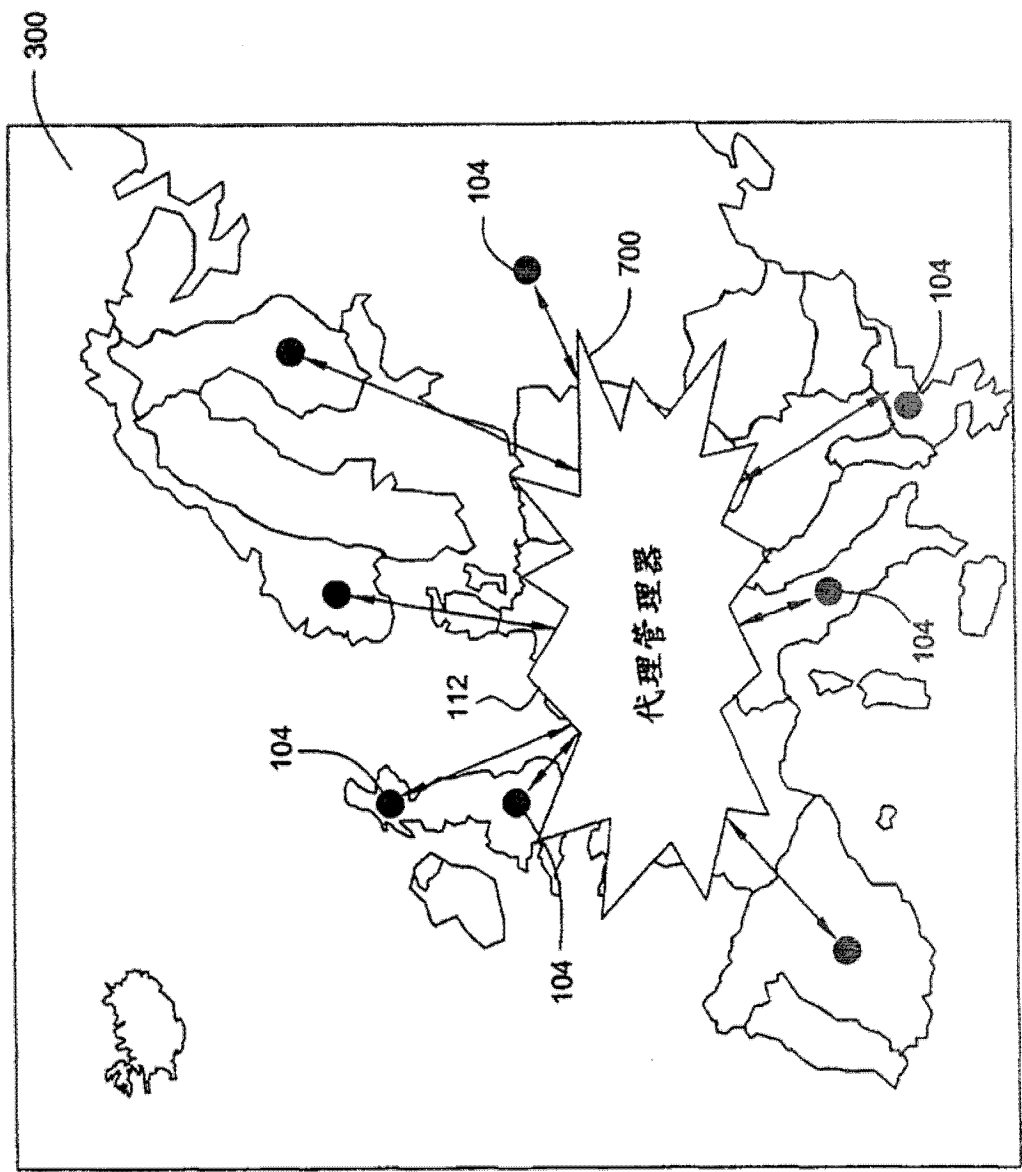


图 4A

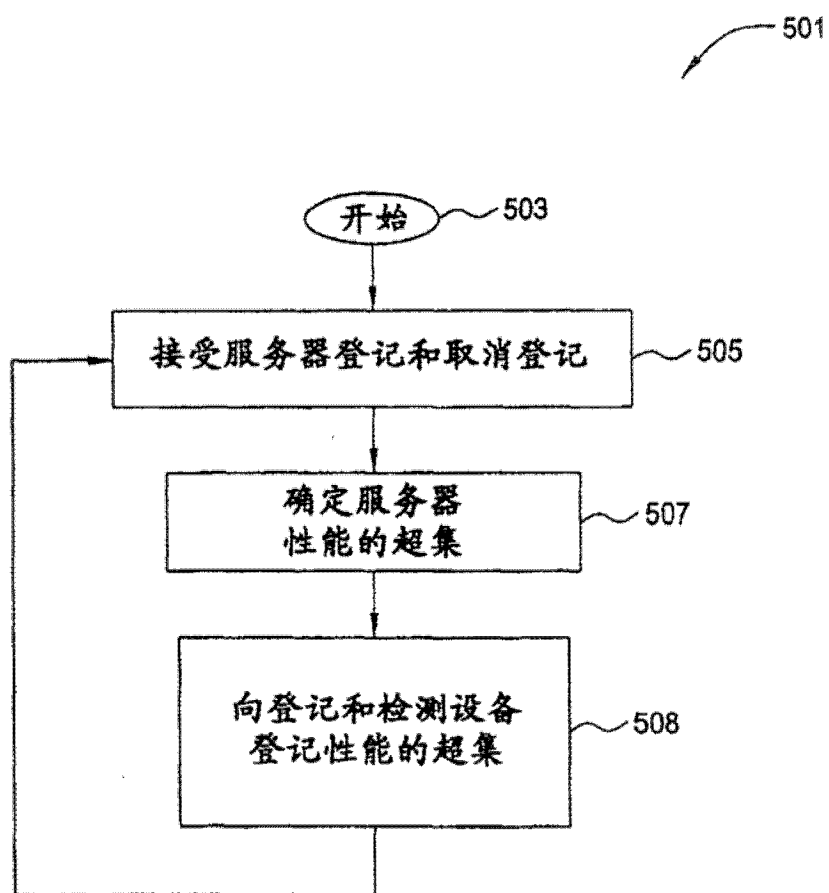


图 4B

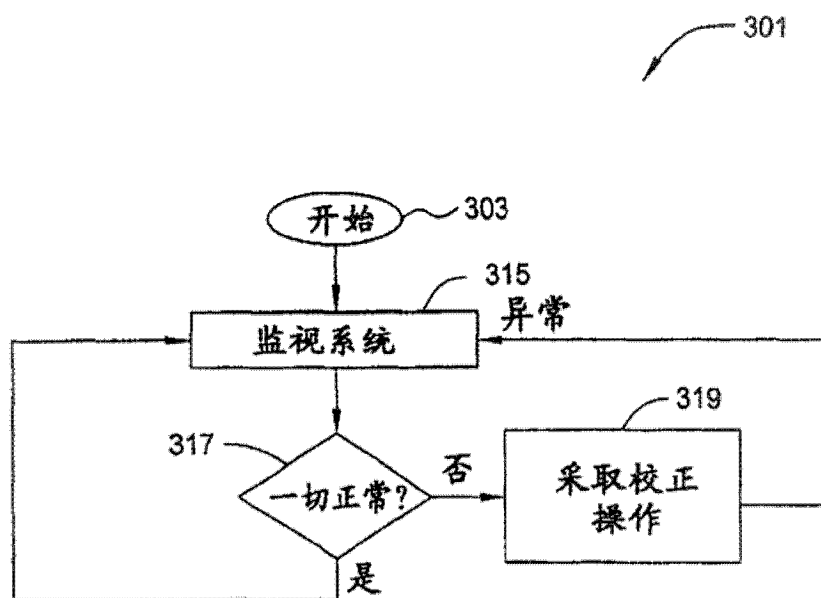


图 4C

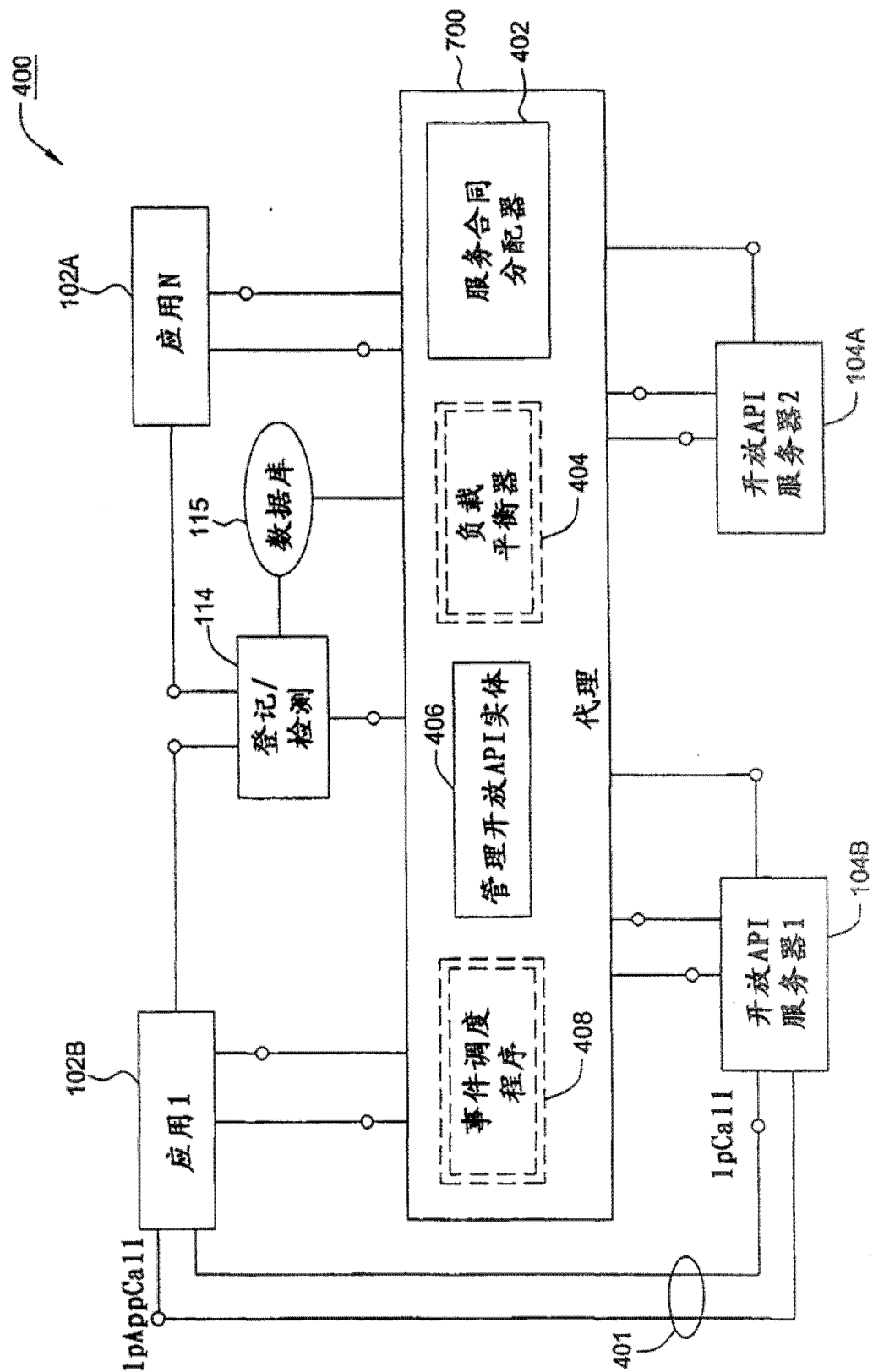


图 5