



[12] 发明专利申请公开说明书

[21] 申请号 200410032293.8

[43] 公开日 2004年10月6日

[11] 公开号 CN 1534520A

[22] 申请日 2004.3.29

[21] 申请号 200410032293.8

[30] 优先权

[32] 2003.3.28 [33] US [31] 10/402, 031

[71] 申请人 微软公司

地址 美国华盛顿州

[72] 发明人 M·J·皮索 R·M·海华德
C·克雷纳曼 P·Y·-K·吴
J·F·诺斯 A·W·史密斯
O·特鲁特纳 F·M·瓦斯

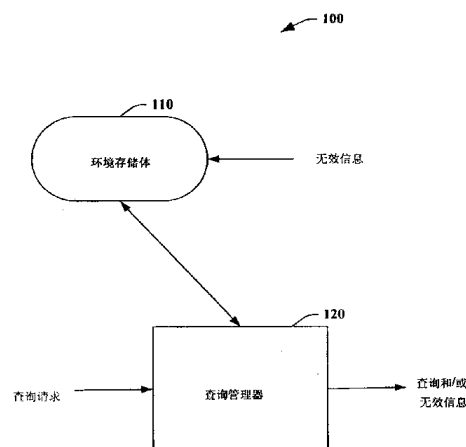
[74] 专利代理机构 上海专利商标事务所
代理人 李家麟

权利要求书6页 说明书17页 附图13页

[54] 发明名称 对数据库结果及推导目标进行高速缓存和无效的系统和方法

[57] 摘要

本发明涉及把从数据库查询产生的结果保存在内存中用于当前和随后的使用、并且当所保存的结果变得不一致时使它们无效，从而减少了对不一致结果的采用的系统和方法，所述结果如原始数据、对象、可查询数据集合、全部或部分 web 应答。所保存的结果可以被一个或多个组件所采用，并且一般在随后查询会返回基本类似的结果时被使用。因此，所述系统和方法减少了对基本上冗余查询的执行，冗余查询会降低性能并且消耗资源。当发生会影响所保存结果的一致性的数据库变化时，以及/或者当期满时段时，可以发送无效消息，其中采用该消息使所保存的结果无效。因此，所述系统和方法减少了对不一致保存结果的采用。此外，可以采用自动重新查询技术自动地刷新不一致结果以获得一致结果。



1. 一种内存无效注册系统，包括：
环境存储体，用于存储无效消息；
查询管理器，用于把数据库查询及相关的无效信息从所述环境存储体提交到数据库；以及
内存，用于存储基于数据库查询的结果，其中所述相关的无效信息由数据库来维持。
2. 如权利要求 1 所述的系统，其特征在于，所述无效信息包括无效字符串、服务信息和到期时段中的至少一个。
3. 如权利要求 2 所述的系统，其特征在于，所述无效字符串包括唯一标识符、机器名、端口、地址和队列名中的至少一个。
4. 如权利要求 3 所述的系统，其特征在于，所述无效字符串还包括验证、加密和协议信息中的至少一个。
5. 如权利要求 1 所述的系统，其特征在于，所述查询管理器与环境存储体一起工作，以便获得与查询一起提交的无效信息。
6. 如权利要求 1 所述的系统，其特征在于，与所存储结果相关的无效信息与多个查询相关联。
7. 如权利要求 1 所述的系统，其特征在于，在如无效消息中定义的会影响所存储结果的数据库变化、以及时间期满中的至少一个以后，所述查询管理器接收无效消息。
8. 如权利要求 7 所述的系统，其特征在于，所述数据库变化被阻止，直到所述变化被同步传递为止。

9. 如权利要求 7 所述的系统, 其特征在于, 所述数据库变化在被异步传递时完成。

10. 如权利要求 1 所述的系统, 其特征在于, 在接收到所述无效消息后自动地使内存中保存的结果无效。

11. 如权利要求 1 所述的系统, 其特征在于, 在接收到所述无效消息后自动地重新查询数据库以更新内存中保存的结果。

12. 如权利要求 3 所述的系统, 其特征在于, 所述唯一标识符在运行时间产生, 用于定义所保存结果与源查询之间的关联。

13. 如权利要求 12 所述的系统, 其特征在于, 所述唯一标识符用于把一个或多个查询与一个保存的结果成组。

14. 如权利要求 12 所述的系统, 其特征在于, 所述唯一标识符被用作安全性机制来减少虚假无效信息和虚假无效消息中的至少一个。

15. 如权利要求 1 所述的系统, 其特征在于还包括用于协调结果存储和无效的内存管理器。

16. 如权利要求 10 所述的系统, 其特征在于, 所述自动无效包括以下几个中的至少一个: 从内存中删除不一致结果; 覆写内存中的不一致结果; 以及标记内存中的不一致结果。

17. 如权利要求 1 所述的系统, 其特征在于, 所述内存包括高速缓存。

18. 如权利要求 1 所述的系统, 其特征在于, 所述内存由多个组件所共享。

19. 如权利要求 1 所述的系统, 其特征在于, 它是基于 SQL 的环境所采用的。

20. 如权利要求 1 所述的系统，其特征在于还包括注释和页面指示中的至少一个，用于请求把所述结果存储在内存中由随后请求所使用，并且当所存储的结果变得与数据库不一致时请求使结果无效。

21. 如权利要求 1 所述的系统，其特征在于，所述环境存储体便于无效信息到查询的传递所述查询，用于产生所存储的结果。

22. 如权利要求 1 所述的系统，其特征在于，所述所存储的结果包括以下中的至少一个：原始数据、经变换的数据、经去标准化的数据、集合数据、总计数据、透视表数据、从原始数据产生的对象、部分 web 应答、完全 web 应答以及可查询的数据。

23. 一种自动内存无效系统，包括：
环境，用于存储一个或多个组件的无效信息；
查询管理器，用于提交数据库查询和无效信息；以及
一个或多个内存，用于存储从数据库查询产生的结果。

24. 如权利要求 23 所述的系统，其特征在于，所述一个或多个内存包括：本地高速缓存内存、辅助高速缓存内存以及远程高速内存。

25. 如权利要求 23 所述的系统，其特征在于还包括安全层，所述安全层与查询引擎和安全性组件一起工作，用于减少用虚假无效信息和虚假无效消息中的至少一个进行的恶意查询。

26. 如权利要求 23 所述的系统，其特征在于还包括一种机制，所述机制用于提供以下的至少一种：当所保存的结果变得不一致时自动使所保存的结果无效；以及通知组件所保存的结果已经变得不一致。

27. 如权利要求 23 所述的系统，其特征在于还包括一个防火墙作为安全层。

28. 如权利要求 27 所述的系统，其特征在于，所述一个或多个内存驻留在所述防火墙中。

29. 如权利要求 27 所述的系统，其特征在于，所述防火墙便于一个或多个内存中的结果存储、结果访问和结果无效中的至少一种。

30. 如权利要求 27 所述的系统，其特征在于，所述一个或多个内存可以同时被所述一个或多个组件所访问。

31. 一种保存已注册结果的方法，所述结果变得不一致时自动地无效，所述方法包括：

把无效信息存储在环境中；
与相关查询一起发送所述无效信息；以及
把从查询产生的结果保存在内存中。

32. 如权利要求 31 所述的方法，其特征在于还包括：在查询后但是在存储结果之前检验被查询的数据库是否有变化，以便确保一致的结果。

33. 如权利要求 31 所述的方法，其特征在于，所述无效信息包括无效字符串、服务信息和到期时段中的至少一个。

34. 如权利要求 33 所述的方法，其特征在于，所述无效字符串包括标识符、机器名、端口、地址和队列名中的至少一个。

35. 如权利要求 34 所述的方法，其特征在于，所述无效字符串还包括验证、加密和协议信息中的至少一个。

36. 如权利要求 31 所述的方法，其特征在于，所述内存是高速缓存内存。

37. 一种当从数据库查询产生的结果变得不一致时自动地使所保存的结果无效的方法，包括：

接收一无效消息，所述无效消息包括无效信息的至少一部分，以及使用所述至少一部分无效信息使所保存的结果无效。

38. 如权利要求 37 所述的方法，其特征在于还包括：自动地使结果无效。

39. 如权利要求 37 所述的方法，其特征在于还包括：自动地重新查询以产生并保存一致结果。

40. 如权利要求 37 所述的方法，其特征在于，所述无效消息与会影响所存储结果的数据库变化以及期满时段中的至少一个相关联。

41. 在两个或多个计算机组件间发送的数据分组，所述数据分组便于把结果存储在内存中并且当所存储的结果变得不一致时自动地使所存储的结果无效，所述数据分组包括：

存储在环境中的无效信息，所述无效信息可以用查询来检索并发送，并且在无效消息中用来自动地使所保存的结果无效。

42. 一种用于存储内存无效和注册系统的计算机可执行组件的计算机可读媒介，所述内存无效和注册系统用于保存结果并且在结果变得不一致时使结果无效，所述媒介包括：

环境，用于存储与组件相关的无效信息；

查询管理器，用于从环境获得无效信息并且与所请求的查询一起发送所述无效信息；以及

内存管理器，用于便于保存结果并且使不一致的结果无效。

43. 一种用于使内存中保存的结果无效的系统，包括：

获得请求查询数据库的组件的无效信息的装置；

与查询一起发送无效信息的装置；

用于接收无效消息的装置；以及

用于使内存中存储的不一致结果无效的装置。

44. 如权利要求 43 所述的系统, 其特征在于还包括自动地重新查询数据库是否有一致结果的装置。

45. 如权利要求 43 所述的系统, 其特征在于还包括验证无效信息和无效消息不虚假的装置。

对数据库结果及推导目标进行 高速缓存和无效的系统和方法

技术领域

本发明一般涉及数据库，尤其涉及为随后的使用而保存从数据库产生的结果、并且当它们变得与数据库不一致时无效所保存的结果的系统和方法。

背景技术

计算和联网技术已经改变了每天生活的许多重要方面。计算机已经成为家用的主要产品，而不是奢侈的教育工具和/或娱乐中心，并且向用户提供了工具来管理并预报财务；控制像取暖、制冷、照明和安全之类的家用操作；并且把记录和图像存储在永久且可靠的媒介中。因特网这样的联网技术向用户提供了对远程系统、信息和相关应用的实际上无限的存取。

随着计算和联网技术变得稳健、安全和可靠，更多的用户、批发商、零售商、企业家、教育机构等等改变了范例，并且采用因特网而不是用传统手段来执行商务。例如，许多商务和用户提供了网站和/或在线服务。例如，现在用户可以通过因特网来访问他/她的账户，并且执行越来越多的有效交易，比如余额查询、基金转账以及账单支付。

一般而言，因特网会话包括用户与客户应用程序(如 web 服务器)连接而与数据库服务器相互作用，数据库服务器把信息存储在客户应用程序可以访问的数据库中。例如，股票市场网站可以向用户提供用于检索股票报价并且购买股票的工具。用户可以输入股票符号并且通过点击鼠标来激活查询而请求股票报价。然后，客户应用程序查询数据库有无股票信息并且返回股票报价。同样，用户可以通过提供适当的信息来购买股票，其中提交订购会促使数据库查询范围当前的价格信息和订购状态。

随着更多的用户利用这种服务的优点，也消耗了更多带宽，这会降低性能和/或速度，因为带宽是一项有限的资源。另一个缺点是与有效数据数量有关的有限的有效数据传输速率。例如，检索大量数据(如，分布在各个服务器中)的请求可能是

时间密集且性能有限的，即使当带宽可用时。

通过用户执行频繁且类似的请求而合成了上述内容。例如，希望当前股票报价的用户可以执行频繁的请求来获得更新的报价。因此，用户频繁地消耗带宽来检索基本类似的信息。此外，其他用户可能同时执行检索基本类似信息的请求。用户还受到限制的是，由于所返回的结果在随后的数据库表变化后会变得无效，因此不能保证所检索的数据的一致性。

发明内容

下面给出了本发明的简要概述，以便提供对本发明某些方面的基本理解。这个概述不是本发明的广泛综述。它并不意图标识出本发明的关键/决定性的元件，或者叙述本发明的范围。它唯一的目的是以简化形式给出本发明的某些概念，作为稍后给出的更详细的描述的序言。

本发明涉及当从数据库查询产生的所保存结果由于数据库变化以及/或者由于期满时段而变得无效时用于对所述结果进行无效的系统和方法。因此，本发明提供了一种机制，用于查询数据库、保存从数据库查询产生的结果、并且使用所保存的结果直到数据库变化和/或到期时段使结果无效。

一般而言，数据库存储可以通过查询数据库而检索到的信息。用户可以与应用程序对接，例如为了获得数据库内存储的至少一部分信息。例如，用户可以采用web服务或者提交与服务器上存储的信息相关的请求。请求会开始有关数据库的一个或多个查询，然后可以使用所述一个或多个查询的结果来产生一个应答，该应答要从该web服务被返回、被显示、被消耗并且/或者进一步被采用。

在许多情况下，数据库会成为资源争用的共用源。例如，不止一个用户会发出一个类似请求，该请求同时查询数据库并且返回类似的结果。这种同时查询类似结果会增加数据库负载、减少带宽、降低性能并且消耗资源。在另一示例中，一个或多个用户会频繁地执行返回类似结果的请求。同样，系统性能会经由如数据库负载而被降低。在上述内容之内，数据库表示在多个用户请求和/或用户间共享的状态，而共享的状态争用会限制系统性能和可伸缩性。

为减少查询数而一般采用的技术包括：把频繁使用的数据（例如，形式为原始数据、经变换的数据、以及可查询形式的数据）保存在数据库外部，并且使用所保存的结果而不是执行另外的查询。然而，由于可能发生随后的数据库变化来提供与数据库不一致的所保存数据而不通知用户，因此所保存的数据的一致性受到损害。

本发明的系统和方法减少了数据库负载(如减少了对基本类似的信息的查询)、提高了可伸缩性、减少了重复处理、并且确保了数据一致性。所述系统和方法包括把无效信息存储在环境存储体(context bank)内,其中可以根据查询来检索并发送所述无效信息,以便把该查询与查询请求者相关联并且在数据库内注册该查询。从查询产生的结果(如,原始数据、经变换的数据、可查询形式的数据、部分应答和全部应答)可以被保存在数据库外部的可存取存储媒介中,并且被各个用户(如 web 服务器)使用一次或多次,从而缓和了对基本类似的信息的查询。

如果发生会影响所存储结果的数据库变化,则可以发送无效消息以便于使所保存的结果无效,从而减少了用户采用不一致数据的情况。一般使用根据查询发出的无效信息来构成无效消息。此外,可以定义一个到期时段,使得在到期时段期满后所保存的结果会被无效,即使不发生数据库变化。在使所保存的结果无效后,例如,这些结果可以被自动地删除并且/或者自动地刷新。

而且,本发明的系统和方法可以在 web 环境中被采用,其中当可以全部或部分地存储从数据库查询产生的 web 应答并且响应于随后的请求而返回这些应答时,可以实现 web 服务器资源的等待时间和使用的进一步减少。例如,开发者可以通过在 web 页面内嵌入注释来请求基于数据库查询产生的应答会被高速缓存,直到查询结果变化为止。注释调用 web 服务器把数据库相关性与要被高速缓存的应答相关联,并且把与相关性相关的无效消息以及/或者相关性自身存储在环境存储体中。根据到数据库的随后查询而自动地检索并发出所存储的无效信息。如上所述,当数据库发生变化提供了不一致的前面查询的结果时,以及/或者当到期时段期满时,可以在无效消息内采用该无效信息。

把无效信息存储在环境存储体中并且使用该信息可以便于例示把无效信息与查询相关联并且保存(如高速缓存)查询应答的基础结构。因此,采用所存储的无效信息减轻了对 web 服务器(和/或能处理查询请求的其他组件)获悉应答产生的要求。而且,采用所存储的无效信息减轻了对产生应答的组件必须获悉应答怎样被 web 服务器使用的要求。

按照发明主题,这里所述的系统和方法采用组件来存储无效信息、执行查询、保存基于查询的结果、接收无效消息以及/或者使所保存的信息无效。此外,可以采用用于异步和同步通信的各种数据传输技术以及安全性技术,这些技术减少了恶意和虚假信息 and 消息的存在和/或传输。实践中,可以在各种环境中采用所述系统和方法,这些环境包括客户机-服务器和/或基于 SQL 的环境。

为了实现上述及相关的目标，本发明包括下面全面描述且在权利要求中特别指出的特征。下面的描述和附图详细提出了本发明的某些说明性方面和实施方式。然而，这些表示出其中可采用本发明原理的各种方式的一小部分。在参照附图考虑了本发明的下列详细描述后，本发明的其他目的、优点和新颖性特征将变得更为明显。

附图说明

- 图 1 说明了按照本发明一方面的示例性无效注册系统。
- 图 2 说明了按照本发明一方面的示例性无效内存管理系统。
- 图 3 说明了按照本发明一方面的示例性数据库注册和无效系统。
- 图 4 说明了按照本发明一方面的示例性基于 web 的内存无效系统。
- 图 5 说明了按照本发明一方面的示例性安全性技术。
- 图 6 说明了按照本发明一方面的示例性无效消息传递技术。
- 图 7 说明了按照本发明一方面的示例性分布式无效系统。
- 图 8 说明了按照本发明一方面的示例性 web 服务无效系统。
- 图 9 说明了按照本发明一方面的示例性自动无效方法。
- 图 10 说明了按照本发明一方面的图 9 的继续。
- 图 11 说明了按照本发明一方面的示例性的基于变化的无效方法。
- 图 12 说明了按照本发明一方面的示例性的基于时间的无效方法。
- 图 13 说明了按照本发明一方面的示例性操作系统。

具体实施方式

现在参照附图描述本发明，附图中使用相同的数字来表示相同的元件。在下列描述中，为了说明目的，提出了许多特定的细节以便提供对本发明的全面理解。然而显然本发明可以没有这些特定细节而实现。在其他情况下，为了便于描述本发明，以框图形式示出公知的结构和设备。

如本申请中所使用的，术语“组件”是指计算机相关的实体，或是硬件、硬件软件的组合、软件，或是执行中的软件。例如，组件可以是、但不限于是：运行在处理器上的进程、处理器、对象、可执行的执行线程、程序和/或计算机。通过说明，运行在服务器上的应用程序以及服务器都可以是计算机组件。一个或多个组件可以驻留在进程和/或执行线程内，组件可以被定位在一个计算机上以及/或者被

分布在两个或多个计算机之间。“线程”是进程内的实体，操作系统内核调度它用来执行。每个线程都有一个相关的“环境”，所述“环境”是与线程执行相关的易失性数据。线程的环境包括系统注册表的内容以及属于线程的进程的虚拟地址。因此，包括线程环境的实际数据在它执行时而改变。

本发明涉及用于把从数据库查询产生的结果存储在(如动态的和静态的)内存中的系统和方法，其中所述结果可以被检索和使用，并且在它们变得不一致时(如数据库变化和到期时段期满)被无效。所述系统和方法通过查询基本相似的(如冗余的)信息而减少了系统负载和资源争用。此外，所述系统和方法减少了不一致的所保存结果的采用。因此，所述系统和方法提供了一种机制，用于查询数据库有无信息；把结果存储在像高速缓存这样的内存中；当期望基本相似的结果时采用所保存的结果一次或多次；当所保存的结果变得不一致时自动使其无效；以及通过自动地重新查询和保存任选地刷新内存。

参照图 1，说明了按照本发明一方面的示例性无效注册系统 100。无效注册系统 100 包括环境存储体 110 和查询管理器 120。

环境存储体 110 可用于提供信息以便于在发生的数据库变化会影响所保存的结果以及/或者当到期时段期满时，使所保存的结果无效。例如，环境存储体 110 可用于存储与组件相关的无效信息(如下述的相关性)。无效信息可以通过数据库查询链接到数据库。结果可以被保存和使用一次或多次。当数据库发生的变化会影响查询以及/或者持续时间期满时，可以使用相应的无效信息以便于无效所保存的结果，例如通过通知、事件、标志、所保存结果的自动无效、以及自动重新查询，从而用一致结果获得并刷新内存。

一般的无效信息包括无效字符串、服务信息和超时。无效字符串一般包括组件标识符、位置(如机器名)、端口、地址和/或队列名，并且还可以包括如验证信息、加密和协议选项。服务信息可以包括标志，这些标志用于自动删除所保存的结果、自动刷新结果、自动把结果标记为无效、以及通知组件而不影响所保存的结果，等等。超时可以基于数据库变化、文件变化和持续时间等等。可以理解，上述内容是说明性的而不是限制性的。因此，按照本发明一方面可以包括附加的、不同的和/或较少的信息。

环境存储体 110 内存储的无效信息可用于查询管理器 120。查询管理器 120 可以在执行命令和/或查询时使用该无效信息。例如，查询管理器 120 可以从环境存储体 110 获得无效信息，并且包括根据查询的无效信息，其中无效信息可以是查

询的一部分和/或与查询相关联。

在本发明一方面中，可以采用页面指示(如 web 页面内的注释)来开始组件的例示(如相关性对象)以及把与组件相关的无效信息存储在环境存储体 110 中。采用环境存储体 110 来存储无效信息便于建立根据查询采用无效信息的基础结构，而无须使调用查询管理器 120 的应用程序必须获悉应答怎样被高速缓存。

当通过请求查询数据库来调用查询管理器 120 时，查询管理器 120 与环境存储体 110 相互作用以检索与组件相关的无效信息。然后，查询管理器 120 通过查询数据库来执行请求，其中从环境存储体 110 检索的无效信息被连续发送并且/或者与查询同时发出。然后返回并使用查询结果以产生要被存储的结果，所发出的无效信息保持与组件、查询和被查询的数据库相关联。

如果被查询的数据库在查询以后并且在产生要被存储的结果(如果要被存储的结果与数据库一致)以前不发生变化，则可以使用结果并将其存储在内存中用于进一步的使用。当随后提交会返回基本相似结果的请求时，可以使用所存储的结果而不是执行另一次查询。因此，本发明减少了对基本相似结果的查询，其中查询基本相似结果会降低性能、吞吐量、处理和速度，并且易受资源争用的影响。

在保存结果之后，数据库变化或时间期满可以反映所保存的结果不一致。当发生的数据库变化会影响上述查询和/或超时期满的结果时，可以采用根据查询发出的无效信息来通知组件(如内存管理器 120)使所保存的结果无效(如删除和刷新)，即使产生该结果的组件不再存在。可以用指示所保存的结果无效的通知来提供试图与所保存的结果相关的其他组件。因此，本发明通过通知组件以及/或者使所保存的结果无效而减少了对不一致保存结果的采用。

参照图 2，说明了按照本发明一方面的示例性无效内存管理系统 200。无效内存管理系统 200 包括查询管理器 120 和内存管理器 220。

除了上述功能以外，查询管理器 120 可以为变化和/或期满的数据库接收查询结果和/或无效信息。一般而言，在根据查询发出无效信息(如上所述)之后，返回查询结果。在本发明一方面中，查询管理器 120 接收所返回的查询结果，并且采用内存管理器 120 来便于保存和/或使用从查询结果产生的结果。如果认识到在产生要被存储的结果前数据库已变化以及/或者到期时段期满，内存管理器 220 就可以把结果提供给组件而无须把结果保存到内存。如果数据是一致的(如没有数据库变化)且到期时段尚未期满，内存管理器 220 就可以连续地和/或同时地把结果提供给组件以及/或者保存到内存。当请求基本类似的查询时可以由组件来访问所保存的

结果,从而减少了基本类似查询的执行,其中执行基本类似的查询会消耗资源并且降低性能。

应该理解,所返回的结果可以包括原始数据、经变换的数据(如去标准化的集合、总计、数据透视表、从原始数据产生的对象以及/或者部分或全部的 web 应答)、以及/或者可查询的数据(如可以通过随后查询而被使用的数据集合,用于获得其子集)。而且应该理解,可以采用各种技术把结果存储到内存。例如,在第一种方法中,可以保存基本上所有的所返回结果(如全部的 web 应答)。在第二种方法中,把一部分所返回的结果(如部分页面)保存到内存。例如,可以保存对于随后请求常见的一部分。在另一例中,保存对于多个请求共用的结果。然后,对从内存检索的共用结果的本地查询可以返回与请求相关的数据。在第三种方法中,可以通过查询或某些其他手段获得个人化的结果,共用结果可以被保存到内存并且被使用。应该理解,上述示例性说明为了说明性目的而提供,而不限制本发明。

在接收到无效消息后,查询管理器 120 便于把数据库变化通知路由到组件并且/或者使所保存的结果无效。在本发明一方面,查询管理器 120 通知组件,组件决定下一个动作。例如,组件可以采用内存管理器 220 使所保存的结果无效。或者,查询管理器 120 采用内存管理器 220,其中内存管理器 220 可以使所保存的结果无效并且/或者通知组件。应该理解,当所保存的结果被无效时,开始请求和相应的保存的组件不再存在。

现在参照图 3,按照本发明一方面说明了示例性的数据库注册和无效系统 300。数据库注册和无效系统 300 包括查询管理器 120、内存管理器 220、任选的客户机 310 和内存 320。应该理解,可以采用不止一个内存管理器 220、客户机 310 和内存 320;然而为了简洁,下面描述了一个客户机实例。

如上所述,环境存储体 110 可以为组件存储无效信息(如上所述),当发生的数据库变化会影响所保存的结果时,以及/或者当到期时段期满时,可以采用该无效信息来提供使所保存结果无效的信息。在本发明一方面,客户机 310 可以向环境存储体 110 提供无效信息—标识符、机器名、端口、地址、数据库名、自动无效不一致的保存结果的指示、自动刷新不一致结果的指示、以及超时(如,根据数据库变化、文件变化和持续时间而无效)。

当客户机 310 调用查询管理器 210 来查询数据库时,查询管理器 120 从环境存储体 110 接收相关的无效信息(如上所述),以及/或者从环境存储体 110 检索相关的无效信息(未示出)。查询管理器 120 接着进行查询并且提供无效信息。应该理

解，查询管理器 120 可以按照本发明一方面来查询动态的和/或静态的数据库。

查询管理器 120 可以接受查询结果，并把结果发送到客户机 310。应该理解，本发明提供了同步和异步的消息传递。在同步消息传递中，可以阻止数据库变化，直到传递无效消息和/或使不一致结果无效为止。同步传递通过确保在提交变化和接收无效消息之间不发生数据库变化，以及/或者使结果无效，从而减少了数据不一致性。在异步消息传递中，在传递无效消息之前以及/或者在使不一致结果无效之前可以提交数据库变化。异步传递减少了传递等待时间。

在接收到查询结果后，接着用各种技术探查被查询的数据库以确认在查询后但在产生要被存储的结果之前是否发生变化。如果发生变化，客户机 310 就可以使用结果而不调用内存管理器 220。如果不发生变化，客户机 310 就可以使用结果并且采用内存管理器 220 来把结果存储在内存 320 中(如本地、远程、高速、HTTP 和共享的内存)。会返回基本相似结果的随后请求可以使用来自内存 320 的所保存的结果，而不是执行查询，直到所保存的结果变得不一致为止。

在本发明另一方面，采用了各种技术以便于确定是否要保存结果。例如，如果确定结果对于多个请求是共用的并且频繁地变化，就保存结果。然而，如果确定结果频繁地变化以及/或者未被频繁使用，且保存成本相对昂贵，就不保存结果。在本发明还有一方面，采用了各种技术来减少内存问题。例如，如果内存 320 是满的，则可以用以下方法来保存结果：先进先出(FIFO)或先进后出(FILO)方法；基于寿命(line 12)的方法，该方法中先删除最旧信息；用途方法，其中先删除最少使用的信息，或者自上一次使用时间最长的信息；基于大小的方法以及基于按键的方法。在另一例中，不保存结果，直到足够的“空闲”内存可用为止。其他方法、或者这些方法的变化会考虑产生结果所需的时间量和/或资源。

一旦数据发生的变化会影响结果以及/或者到期时段期满，就向系统 300 发送一无效消息。一般而言，无效消息包括无效信息的至少一部分。查询管理器 120 可以接收无效消息。然后，查询管理器 120 通知客户机 310(和/或为接收无效消息而注册的任何其他组件)。然后，客户机 310 可以调用内存管理器 220 来使所保存的结果无效(如删除、擦除、覆写和移动)。如果无效信息内指示自动刷新结果，就执行随后的查询来产生新结果，新结果可以被保存在内存 320 中并且被像客户机 310 这样的组件所使用。采用上述技术可以通过在不一致结果上保存新结果而减少对不一致结果的删除。

应该理解，无效注册系统 100、无效内存管理系统 200 以及数据库注册和无效

系统 300 还可以采用组件来注册数据库(如数据库表、数据表和表格)、检测数据库的变化、确定是否注册了已变化的数据库、以及/或者发送指示注册数据库已变化的消息。

尽管图 1-3 是说明系统 100-300 的组件的框图,然而可以理解,环境存储体 110、查询管理器 120、内存管理器 220、客户机 310 和内存 320 可以用一个或多个计算机组件来实现,组件的属于已在此定义。因此应该理解,可以在计算机可读媒介上存储计算机可执行组件,所述计算机可执行组件用于实现系统 100-300、环境存储体 110、查询管理器 120、内存管理器 220、客户机 310 和内存 320,按照本发明,所述计算机可读媒介包括、但不限于:ASIC(专用集成电路)、CD(光盘)、DVD(数字化视频光盘)、ROM(只读存储器)、软盘、硬盘、EEPROM(电可擦除可编程只读存储器)以及存储棒。

参照图 4,说明了按照本发明一方面的示例性的基于 web 的内存无效系统 400。基于 web 的内存无效系统 400 包括 web 服务器 405、相关性组件 410、环境组件 420、查询管理器 430、服务器 440、数据库服务器 450、内存存储器 460、本地内存 470、高速内存 480、其他内存 490(如远程高速内存)、应用程序 492 以及辅助层 494。

Web 服务器 405 可以访问包括页面指示(如上述的注释)的网页。页面指示可用于指出将保存 web 应答,以及在相关数据库变化时和/或当到期时段期满时应该使所保存的应答无效。Web 服务器 405 可以使用页面指示、调用相关性组件 410(如对象或推导的对象)、以及便于把无效信息存储在环境组件 420 中。

相关性组件 410 获得唯一标识符,例如全局唯一的标识符,即 GUID,用于标识要与随后查询相关联的高速缓存的应答。相关性组件 410 向环境组件 420 提供该唯一标识符以及各种其他的无效信息(如上所述)。一般而言,无效信息包括无效字符串(如, GUID、机器名、端口、地址和队列名、以及任选的验证信息、加密和协议选项)、服务信息和超时(如数据库变化和时间期满的超时)。然而,上述仅仅是说明性的,而不限制无效信息。

当调用查询管理器 430(如查询管理器 110)来查询数据库时,查询管理器 430 与环境组件 420 相连,并且获得相关的无效信息。然后,查询引擎 420 把所述查询和无效信息发送到数据库服务器 450 内的服务器 440(如动态的和静态的)。根据返回到应用程序 492 的结果而对数据库 440 执行查询。无效信息留在数据库服务器 450 内,并且提供 web 服务器 405 上的相关性组件 410 以及服务器 440 所返回的结果之间的关联。

在接收到查询结果后，web 服务器产生全部或部分的应答。Web 服务器 405 可以轮询相关性组件 410 以确定在产生应答时是否发生数据库变化，然后决定是否保存结果。如果确定要保存结果，则 web 服务器 405 可以采用内存存储器 460 以便于把结果保存到本地内存(如高速缓存)、高速内存 480(如高速缓存)和/或其他内存 490。一般而言，本地内存 470 可以与应用程序 490 相关联，高速内存 480 可以与辅助层 494(如 HTTP.sys)相关联。否则，可以使用应答以及/或者不保存而删除应答。

当发生的数据库变化会影响所保存的 web 应答时，以及/或者当到期时段期满时，服务器 440 会向相关性组件 410 发送无效消息。然后，相关性组件 410 然后发信号通知该变化(如通过引发一个事件并设定一个标志)。内存存储器 460 接收该信号并且使应答无效。或者，当接收到无效消息时，可以采用自动刷新来执行查询以获得并保存一致结果。自动刷新会覆写一致结果并且/或者在保存一致结果前删除不一致结果。

接着参照图 5，说明了按照本发明一方面的示例性安全性技术。系统 500 包括 web 服务器 405、数据库服务器 450 以及使 web 服务器 496 和数据库服务器 450 操作上耦合的安全层 510。

如上所述，查询引擎 410 与环境组件 420 相连以获得接收查询请求后的无效信息。然后，查询引擎 420 把所述查询和无效信息从 web 服务器 405 发送到数据库服务器 450 内的服务器 440。

前述的无效信息可以包括像全局唯一标识符(即 GUID)这样的唯一标识符。GUID 提供了一种安全性机制，该机制可用于减少对模拟数据库变化和时间段期满的恶意企图。例如，在本发明一方面，在接收到数据库查询注册请求时，在运行时间创建 GUID，从而减少了“猜测”或仿造 GUID 的机会。接着使用 GUID 来标识要被无效的所保存的结果的集合。

如果确定 GUID 无效(如它不标识要被无效的一组保存结果)，则可以与 GUID 一起忽略并/或删除无效请求。在另一例中，虚假的无效请求可以被转发给系统管理员并且用于跟踪始发位置。

图 6 说明了按照本发明一方面的示例性无效消息传递技术。系统 600 包括 web 服务器 405、数据库服务器 450 以及使 web 服务器 405 和数据库服务器 450 操作上耦合的通信接口 610。在接收到查询请求后，查询引擎 430 从环境组件 420 获得无效信息，并把所述查询和无效信息发送到数据库服务器 450。根据被发送到 web 服

务器 405 的结果来执行查询，其中结果可以被保存在内存中并且不止一次地使用，来减少执行基本类似查询的次数。

在结果从数据库服务器 450 返回之后，无效信息留在数据库服务器 450 内，并且当发生数据库变化来把相关的保存结果定位在 web 服务器 405 中时可以采用该无效信息。应该理解，可以采用各种机制把无效信息存储在数据库服务器 450 中，并且可以使用各种技术把无效信息与变化的数据库相匹配并且发送无效消息。

当发生数据库变化时，构成无效消息并且把它在通信接口 610 上从数据库服务器 450 发送到 web 服务器 496。无效消息可以包括无效信息的至少一部分。

应该理解，数据库变化可以被提交以及/或者被保持。例如，在本发明一方面，可以采用异步无效消息传递模型，其中无论是否在通信接口 610 上发送无效消息以及/或者无论所保存的结果是否已被无效，都提交数据库变化。

在本发明另一方面，可以采用同步无效信息传递模型，其中阻止数据库变化，直到在通信接口 610 上发出无效消息为止，以及/或者直到所保存的结果已经被无效为止。异步模型提供伸缩性，而同步模型提供了可靠性。为了便于理解而提供了上述内容，这限制本发明，例如按照本发明也可以采用各种其他的推拉方法(push and/or pull)。

图 7 说明了按照本发明一方面的示例性分布式无效系统 700。无效系统 700 包括：第一 web 应用程序 710₁ 到第 N 个 web 应用程序 710_N，N 是大于等于 1 的整数；第一环境组件 720₁ 到第 M 个环境组件 720_M，M 是大于等于 1 的整数；第一数据库 730₁ 到第 K 个数据库 730_K，K 是大于等于 1 的整数；以及共享内存 740。

第一 web 应用程序 710₁ 到第 N 个 web 应用程序 710_N 可以总称为 web 应用程序 710，第一环境组件 720₁ 到第 M 个环境组件 720_M 可以总称为环境组件 720，第一数据库 730₁ 到第 K 个数据库 730_K 可以总称为数据库 730。

可以采用 web 应用程序 710 (如对象实例) 来查询数据库以满足用户请求。一般而言，在例示了 web 应用程序 710 后，与 web 应用程序 710 相关的无效消息就被汇编并被存储在相应环境组件 720 (如环境存储体 110) 中。无效信息一般包括上述的唯一标识符、无效字符串、服务信息以及到期时段；然而，无效信息不限于此。

当由至少一个 web 应用程序 710 接收到查询请求时，采用相应环境存储体来获得相关的无效信息。然后，通过通信协议 (如以太网和 Firewire 协议) 把查询和无效信息发送到数据库 730。注意到查询可以采用一个或多个数据库 730 来满足用户请求。

在执行查询之后，把结果返回到至少一个 web 应用程序 710。为了确保结果是一致的，可以检验数据库 730 以确定在查询后但在产生要被存储的结果之前是否发生会影响结果的变化。如果确定发生变化，则使用并且/或者删除结果。如果确定结果是一致的，则结果可以被使用并且/或者被保存在内存中，例如共享内存 740。因此，本发明还减少了不止一个 web 应用程序 710 执行对基本类似结果的查询。

当至少一个数据库 720 发生数据库变化时，并且/或者当到期时段期满时，至少一个数据库 720 把无效消息发送到一个或多个 web 应用程序 710 以及/或者共享内存 740。应该理解，(如一个)无效消息可以包括不止一个 web 应用程序地址和/或其他信息(如其他无效消息)，以便通知不止一个 web 应用程序受该变化的影响。

在本发明另一方面，发出不止一个无效消息，其中无效消息与 web 应用程序和/或共享内存 740 相关联并且被发送到后两者。在还有一方面，可以广播无效消息，使得基本上所有 web 应用程序 710 和/或共享内存 740 都能接收无效消息。Web 然后应用程序 710 可以决定是否接受并且/或者作用于无效消息。应该理解，向共享内存 740 提供无效消息减少了必须向 web 应用程序提供无效消息的要求。例如，开始请求的 web 应用程序无须存在以使所保存的结果无效。此外，共享所保存结果的任何应用程序都无须存在以使所保存的结果无效。因此，本发明提供了一种机制，即使当开始应用程序和/或任何共享应用程序不再存在时也删除不一致结果和/或空闲内存。

在接收到无效消息以后，可以接着使所保存的结果无效。例如，可以从内存中删除所保存的结果，或者将其标记为不一致。或者，可以采用自动刷新来执行随后的查询以获得要保存到内存的不一致结果。自动刷新可以进一步与随后查询一起发送无效信息，以便在经刷新的结果变得不一致时自动地使它们无效。

图 8 说明了按照本发明一方面的示例性 web 服务无效系统 800。Web 服务无效系统 800 包括客户机 805、防火墙 810、内存 820、第 1 个 web 服务 830₁ 到第 N 个 web 服务 830_N、N 是大于等于 1 的整数；以及数据库 840。第一 web 服务 830₁ 到第 N 个 web 服务 830_N 可以总称为 web 服务 830。

客户机 805 通过防火墙 810 与 web 服务 830 相连。防火墙 810 可以控制对 web 服务 830 的访问。例如，防火墙 810 可以限制客户机 805 到 web 服务 830 的访问以及/或者拒绝到 web 服务 830 的访问。此外，可以采用防火墙 810 来限制客户机 805 到数据库 840 和内存 820 的访问。应该理解，到 web 服务器 830、数据库 840 和/或内存 820 的访问会是相互排斥的。例如，客户机 805 可以对 web 服务器 830 完全

访问，而不访问数据库 840 和/或内存 820。然而，客户机 805 可以对 web 服务器 830、数据库 840 和/或内存 820 有完全访问或者不访问。

还可以采用防火墙 810 来限制对数据库 840 和/或内存 820 中保存的结果的 web 服务访问。在本发明一方面，一个或多个 web 服务 830 可以向数据库 840 发送查询(如由客户机 805 开始)、无效信息(如上所述)、以及像路由信息这样的其他信息。如果一个或多个 web 服务 830 具有许可(clearance)，则可以把发出的信息传递至数据库 840。在本发明另一方面，一个或多个 web 服务 830 可以向数据库 840 发送查询、无效信息及其他信息，而不通过防火墙 810 传递信息。

在执行查询后，数据库 840 返回查询结果，查询结果用于产生要被存储的结果。在本发明一方面，结果被返回至防火墙 810，其中防火墙 810 采用一种机制(如内存管理器 120)把这些结果保存在内存 820 中。在本发明另一方面，结果可以被返回至 web 服务 830(如通过防火墙 810 和/或在防火墙 810 周围)。然后，web 服务 830 可以把结果存储在内存 820 中，其中通过访问内存 820，结果对于 web 服务 830 是可用的，然后把结果提供给客户机 805。

所保存的结果可以被 web 服务 830 访问，web 服务 830 具有采用内存 820 的许可。例如，web 服务 830_N 可以发送一查询，该查询会返回与内存 820 中保存的结果的一个子集基本相似的结果。如果 web 服务 830_N 能访问内存 820，web 服务 830_N 就可以使用所保存的结果，而不是执行查询。如果 web 服务 830_N 不能访问内存 820，则执行查询。

当发生会影响内存 820 中保存额结果的数据库变化时，以及/或者当到期时段期满时，数据库 840 可以向防火墙 810 发送无效消息。如果无效消息具有许可，则可以使用该消息使内存 820 中的不一致结果无效。在本发明一方面，可以另外向 web 服务 830 发送一个通知，来通知 web 服务 830 结果已被无效。另一方面，web 服务 830 会周期性地轮询内存 820 以确定结果是否仍然一致，而不是向 web 服务 830 发送通知。另一例中，当 web 服务 830 发送查询和无效信息时，搜索内存 820 以确定类似的结果是否被保存在内存 820 中，以及类似结果是否一致。

可以采用自动刷新来重新查询一致结果并且重新发送无效信息，从而当下一个数据库变化发生时，以及/或者当对应于数据库数据(结果)的时段期满时，进行无效。

图 9-12 说明了按照本发明的方法。为了说明的简洁，用一系列动作来描述这些方法。应该理解，本发明不限于所说明的动作以及/或者动作顺序，例如动作

可以以各种顺序发生以及/或者同时发生,并且具有这里未给出和描述的其他动作。而且,不需要所有所述的动作来实现按照本发明的方法。此外,本领域的技术人员会理解,或者可以用一系列相互关联的状态(如状态图)或事件来表示方法。

参照图 9,说明了示例性的自动无效方法 900。在参考数字 910 处,举例说明了一个实例(如对象、推导的对象、客户机、应用程序、客户应用程序和相关性实例)。在 920 处,与该实例相关的无效信息被存储在环境中。无效信息一般包括无效字符串、服务信息和超时,如上所述。应该理解,可以举例说明不止一个实例,而且不止一个实例可以把无效信息存储在类似环境中。

在参考数字 930 处,该方法等待接收执行查询的请求。一旦接收到一个请求,就检索出被存储在环境中并且与实例相关联的无效信息。然后在 940 处,查询和无效信息被发送到如其中驻留数据库的动态和/或静态服务器。

现在参照图 10,说明了按照本发明一方面的图 9 的继续。参照参考数字 1010,方法 900 等待查询结果。在接收并处理了查询结果后,在 1020 处检验查询中所使用的数据库,以确认在查询和产生可高速缓存的结果的时间之间是否发生变化。这确保了结果是一致的。

如果在查询和接收结果的时间之间未发生变化,则在 1030 内把结果存储在内存中,并且在 1040 中提供给实例。所保存的结果可用于进一步的采用,例如减少对会返回相似结果的查询的执行。如果在 1020 处确定在查询和产生要被存储的结果之间未发生变化,则在 1040 处一般把结果提供给实例并且不保存。然而,实例可以决定存储并使用不一致的结果。可以采用各种技术来通知其他实例:结果是不一致的。例如,只要当实例试图采用不一致的结果时就可以设定一个标志来产生引发事件。

接着在图 11 处,说明了按照本发明一方面的示例性的基于数据变化的无效方法 1100。参照 1100,发生数据库变化,发送并接收无效消息。无效消息一般包括一部分无效信息。例如,可以包括唯一标识符以便于标识要被无效的一组不一致结果。

应该理解,可以使数据库变化与发送/接收同步,其中当接收到和/或作用于无效消息时提交数据库变化。在另一例中,采用异步技术,其中无论是否发生发送以及/或者接收到消息,都提交数据库变化。

在 1120 处,所保存的结果被无效。应该理解,可以采用使结果无效的各种方法。例如,结果可以从内存中被删除、被标记为无效以及/或者被覆写。在 1130

处,可以执行自动刷新来查询当前的且一致的结果,并且为随后的数据库变化重新注册无效消息。

现在参照图 12,说明了按照本发明一方面的另一个示例性的基于时间的无效方法。参照 1210,无效信息内包括的时间段期满。随后,发送并接收无效消息。无效消息可以包括一部分无效信息。例如,可以包括唯一标识符来标识要被无效的一组不一致结果。在 1220 处,所保存的结果被无效(如类似于上面)。在 1210 处,可以执行自动刷新来查询当前的且一致的结果,并且为随后的期满而重新注册无效消息。

为了提供本发明各方面的附加环境,图 13 和下列讨论提供了对其中可实现本发明各方面的适当操作环境 1310 的简要、一般的描述。虽然用计算机可执行指令的一般环境描述了本发明,比如可由一个或多个计算机或其他设备执行的程序模块,然而本领域的技术人员会认识到,本发明也可以与其他程序模块结合实现,并且/或者用硬件和软件的组合来实现。然而一般而言,程序模块包括执行特定任务或实现特定数据类型的例程、程序、对象、组件、数据结构等等。操作环境 1310 仅仅是适当操作环境的一个例子,而不限本发明的用途或功能范围。其他适用于本发明的公知计算机系统、环境和/或配置包括、但不限于:个人计算机、手提或膝上型设备、微处理器系统、基于微处理器的系统、可编程用户电子设备、网络 PC、小型计算机、大型计算机、包括上述系统或设备的分布式计算环境、等等。

参照图 13,用于实现本发明各方面的示例性环境 1310 包括计算机 1312。计算机 1312 包括处理单元 1314、系统内存 1316 和系统总线 1318。系统总线 1318 耦合系统组件,系统组件包括但不限于:把系统内存 1316 耦合到处理单元 1314。处理单元 1314 可以是各种可用处理器的任何一个。双重微处理器及其他多处理器结构也可以用作处理单元 1314。

系统总线 1318 可以使几种总线结构的任一种,包括内存总线或内存控制器、外围总线或外部总线、以及/或者使用多种可用总线结构的本地总线,多种总线结构包括、但不限于:8 位总线、工业标准结构 (ISA)、微通道结构 (MSA)、扩展 ISA (EISA)、智能驱动设备 (IDE)、VESA 本地总线 (VLB)、外围组件互连 (PCI)、通用串行总线 (USB)、高级图形端口 (AGP)、个人电脑内存卡国际互连总线 (PCMCIA) 以及小型计算机系统接口 (SCSI)。

系统内存 1316 包括易失性内存 1320 和非易失性内存 1322。基本输入/输出系统 (BIOS) 被存储在非易失性存储器 1322 中,它包括如启动期间在计算机 1312 内元

件间传递的例程。通过说明但不限制，非易失内存 1312 可以包括：只读存储器 (ROM)、可编程 ROM (PROM)、电可编程 ROM (EPROM)、电可擦除 ROM (EEPROM) 或闪存。易失性内存 1320 可以包括：随机存取存储器 (RAM)，它充当外部高速缓存内存。通过说明但不限制，RAM 在许多形式中可用，如同步 RAM (SRAM)、动态 RAM (DRAM)、同步 DRAM (SDRAM)、双重数据速率 SDRAM (DDR SDRAM)、高级 SDRAM (ESDRAM)、Synchlink DRAM (SLDRAM) 以及直接 Rambus RAM (DRRAM)。

计算机 1312 还包括可移动/不可移动的、易失性/非易失性的计算机存储媒介。图 13 说明了例如磁盘存储器 1324。磁盘存储器 1324 包括、但不限于：像磁盘驱动、软盘驱动、磁带驱动、Jaz 驱动、Zip 驱动、LS-100 驱动、闪存卡或存储棒这样的设备。此外，磁盘存储器 1324 可以包括独立的存储媒介，或者与其他存储媒介组合，其他存储媒介包括、但不限于：像光盘只读设备 (CD-ROM)、CD 可记录驱动器 (CD-R 驱动器)、CD 可写驱动器 (CD-RW 驱动器) 或数字化视频光盘只读驱动器 (DVD-ROM) 这样的光盘驱动器。为了便于把磁盘存储设备 1324 连到系统总线 1318，一般使用像接口 1326 这样的可移动或不可移动的接口。

应该理解，图 13 描述了充当适当操作环境 1310 中描述的用户和基本计算机资源之间的中间者的软件。这种软件包括操作系统 1328。操作系统 1328 可以被存储在磁盘存储器 1324 上，用于控制并分配计算机系统 1312 的资源。系统应用程序 1330 通过操作系统 1328 到程序模块 1332 和程序数据 1334 利用资源管理，上述三者或被存储在系统内存 1316 中，或被存储在磁盘存储器 1324 上。应该理解，本发明可以用各种操作系统或操作系统的组合来实现。

用户通过输入设备 1336 把命令或信息输入到计算机 1312 中。输入设备 1336 包括、但不限于：像鼠标这样的指示设备、轨迹球、记录笔、触摸板、键盘、麦克风、数码相机、web 照相机等等。这些及其他输入设备经由接口端口 1338 通过系统总线 1318 连到处理单元 1314。接口端口 1338 包括例如：串行端口、并行端口、游戏端口和通用串行总线 (USB)。输出设备 1340 使用某些与输入设备 1336 相同类型的端口。因此例如，可以使用 USB 端口向计算机 1312 提供输入，并且从计算机 1312 把信息输出到输出设备。提供输出适配器 1342 来说明在其他输出设备 1340 中存在某些需要特殊适配器的输出设备 1340，像监视器、扬声器和打印机。输出设备 1342 包括、但不限于：在输出设备 1340 和系统总线 1318 之间提供连接手段的显卡和声卡。应该注意到，其他设备和/或设备系统既提供了输入能力又提供了输出能力，比如远程计算机 1344。

计算机 1312 可以在使用到一台或多台远程计算机(比如远程计算机 1344)的逻辑连接的网络化环境中工作。远程计算机 1344 可以是个人计算机、服务器、网络 PC、工作站、基于微处理器的设备、对等设备或其他公共网络节点、等等,并且一般包括关于计算机 1312 的许多或全部元件。为了简洁,与远程计算机 1344 一起说明了内存存储设备 1346。远程计算机 1344 通过网络接口 1348 逻辑上连接到计算机 1312,然后通过通信连接 1350 物理地连接。网络接口 1348 包含像局域网(LAN)和广域网这样的通信网络。LAN 技术包括光纤分布式数据接口(FDDI)、铜线分布式数据接口(CDDI)、以太网/IEEE 802.3、令牌环/IEEE 802.5 等等。WAN 技术包括、但不限于:点对点链路、像集成服务数字网络(ISDN)及其变化这样的电路交换网络、以及数字用户线(DSL)。

通信连接 1350 是指用于把网络接口 1348 连到总线 1318 的硬件/软件。虽然为了说明简洁示出通信连接 1350 在计算机 1312 内,然而它也可以在计算机 1312 外部。连到网络接口 1348 所必要的硬件/软件包括,仅为了示例性目的,内部和外部技术,比如包括常规电话等级调制解调器、电缆调制解调器和 DSL 调制解调器在内的调制解调器、ISDN 适配器以及以太网卡。

上面描述的内容包括本发明的示例。当然它不可能描述为了描述本发明所使用的组件和方法的每个可想得到的组合,但是本领域的普通技术人员可以认识到,本发明的许多进一步组合和交换是可能的。因而,本发明试图包含落在所附权利要求的精神和范围内的所有这样的替代、修改和变化。而且,在某种程度上术语“包括”或用于描述详细描述,或用于描述权利要求,该术语与权利要求中被用作连词的“包括”有类似的包括性作用。

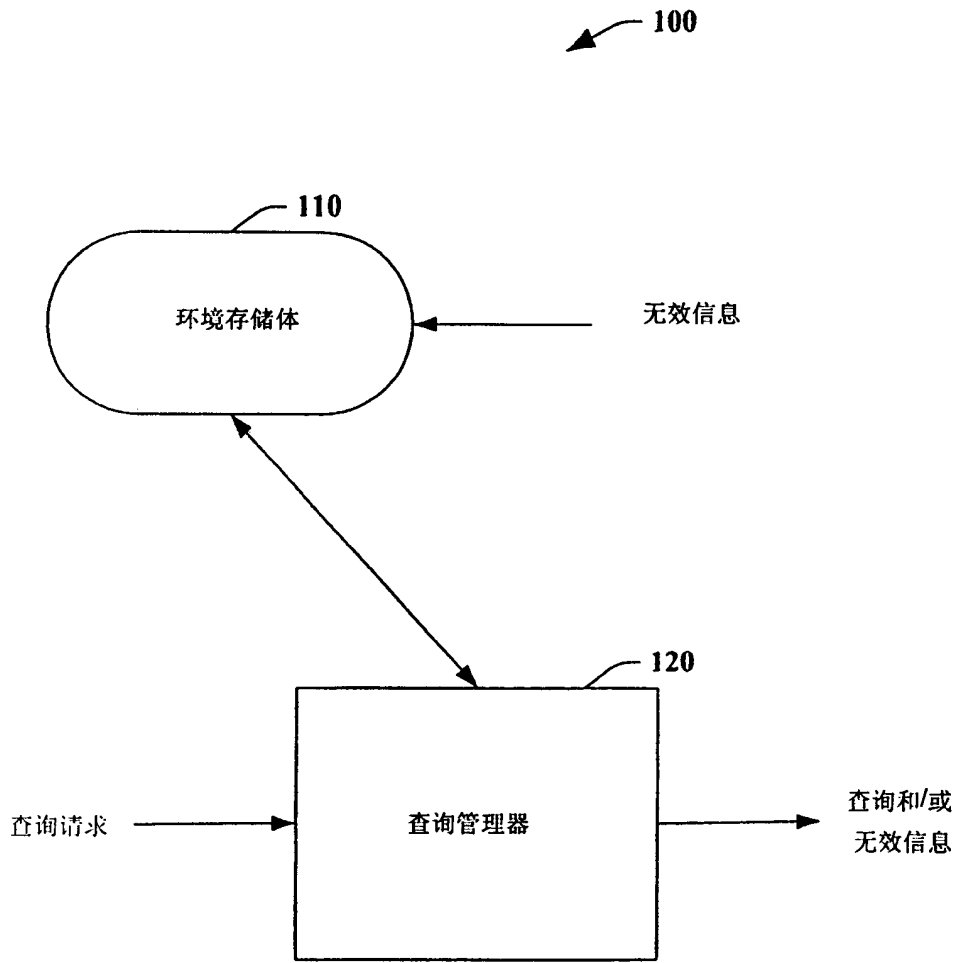


图 1

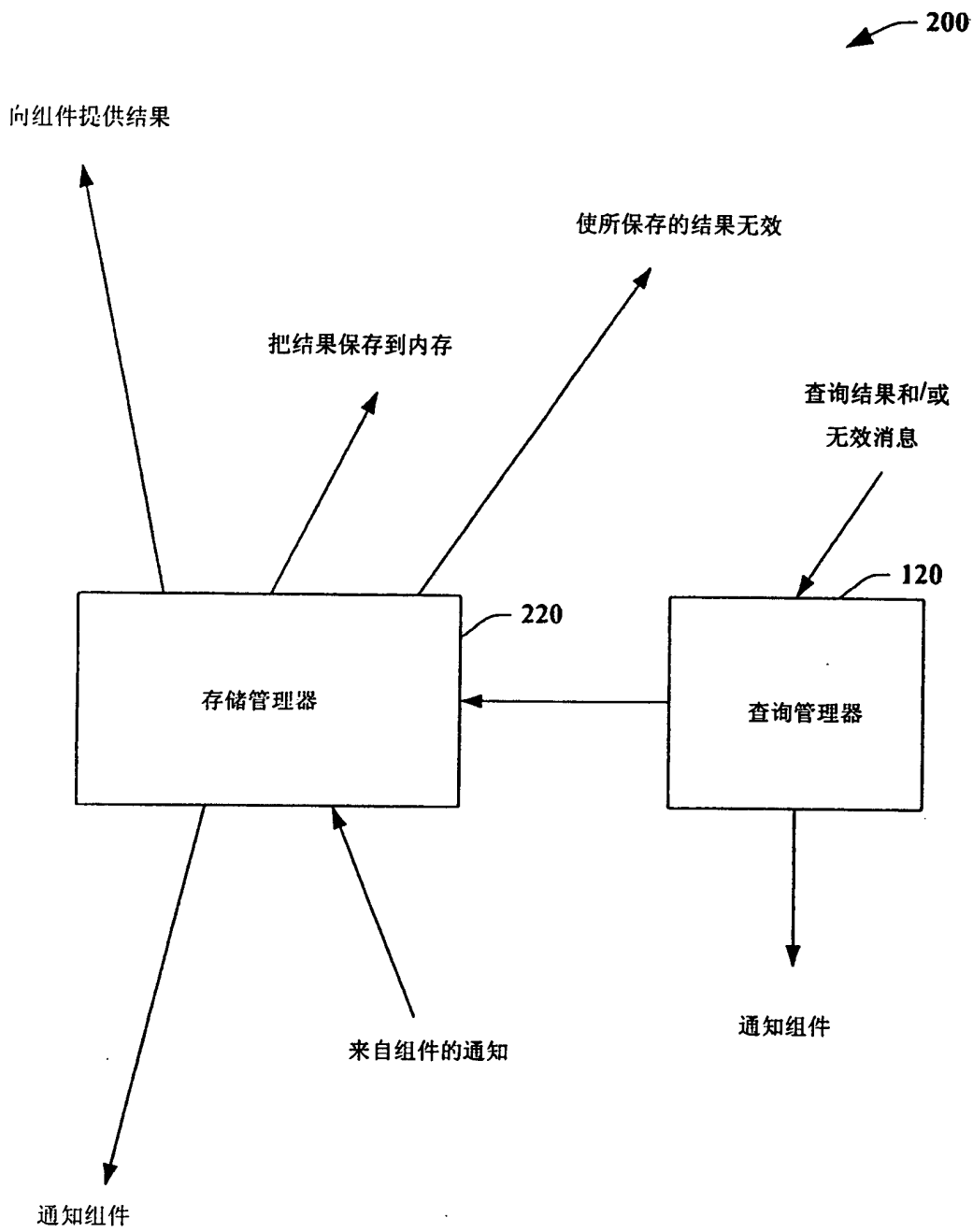


图 2

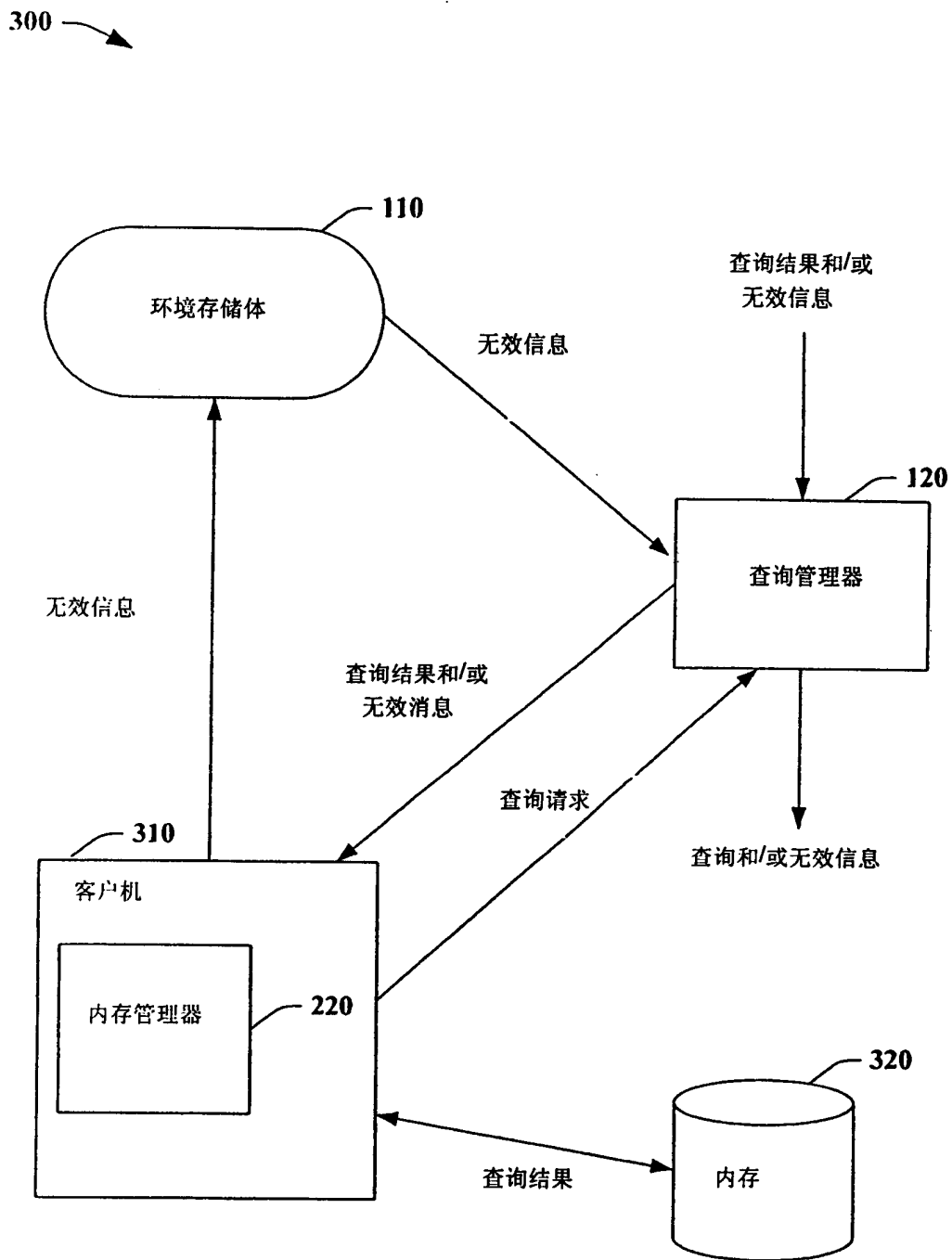


图 3

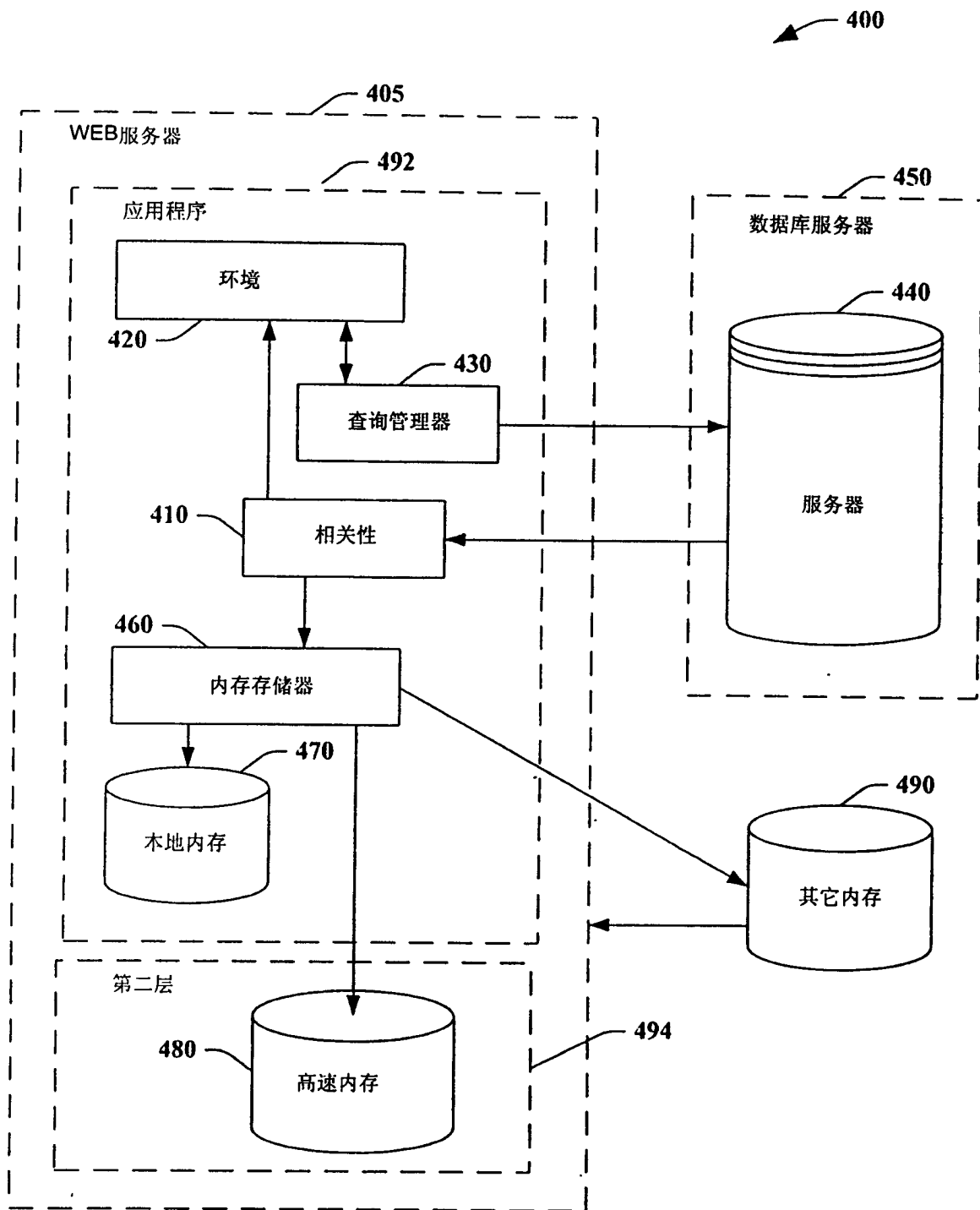


图 4

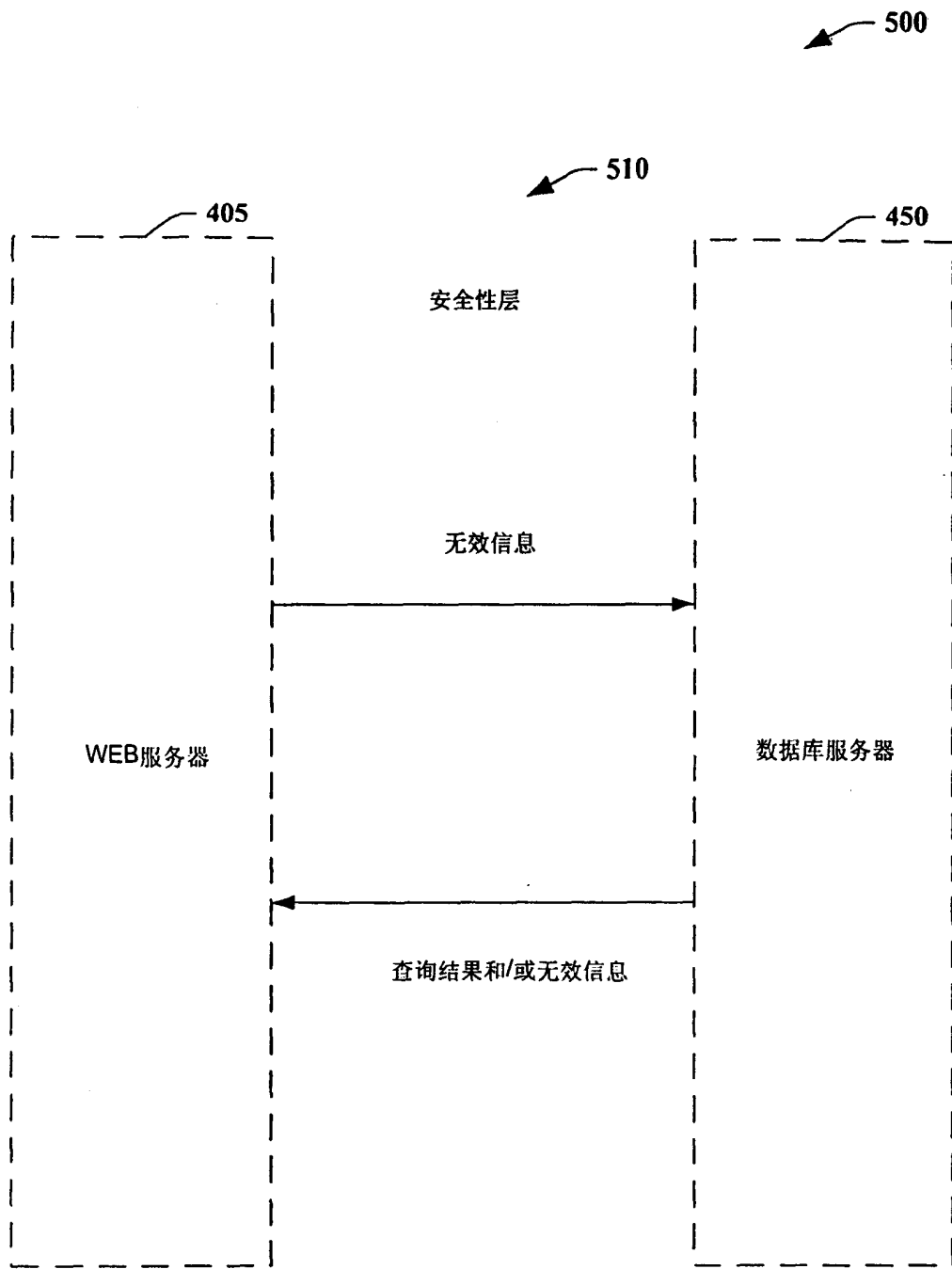


图 5

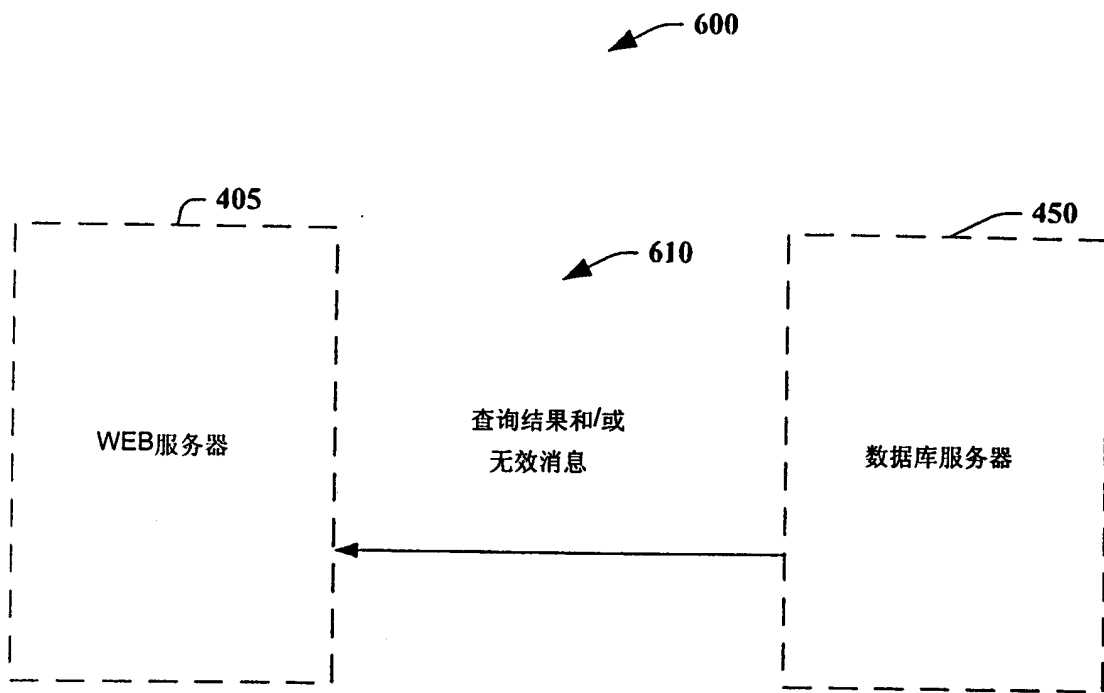


图 6

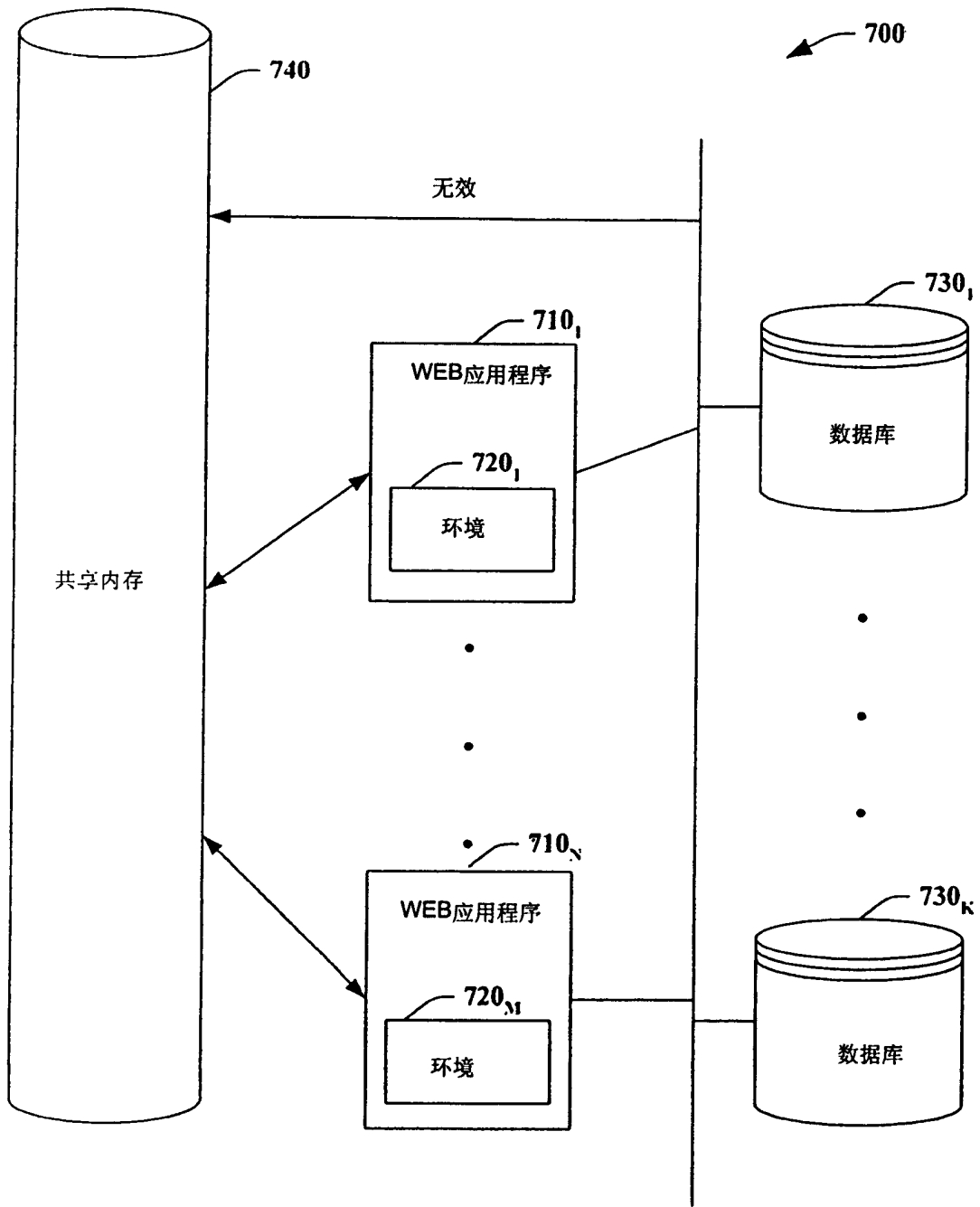


图 7

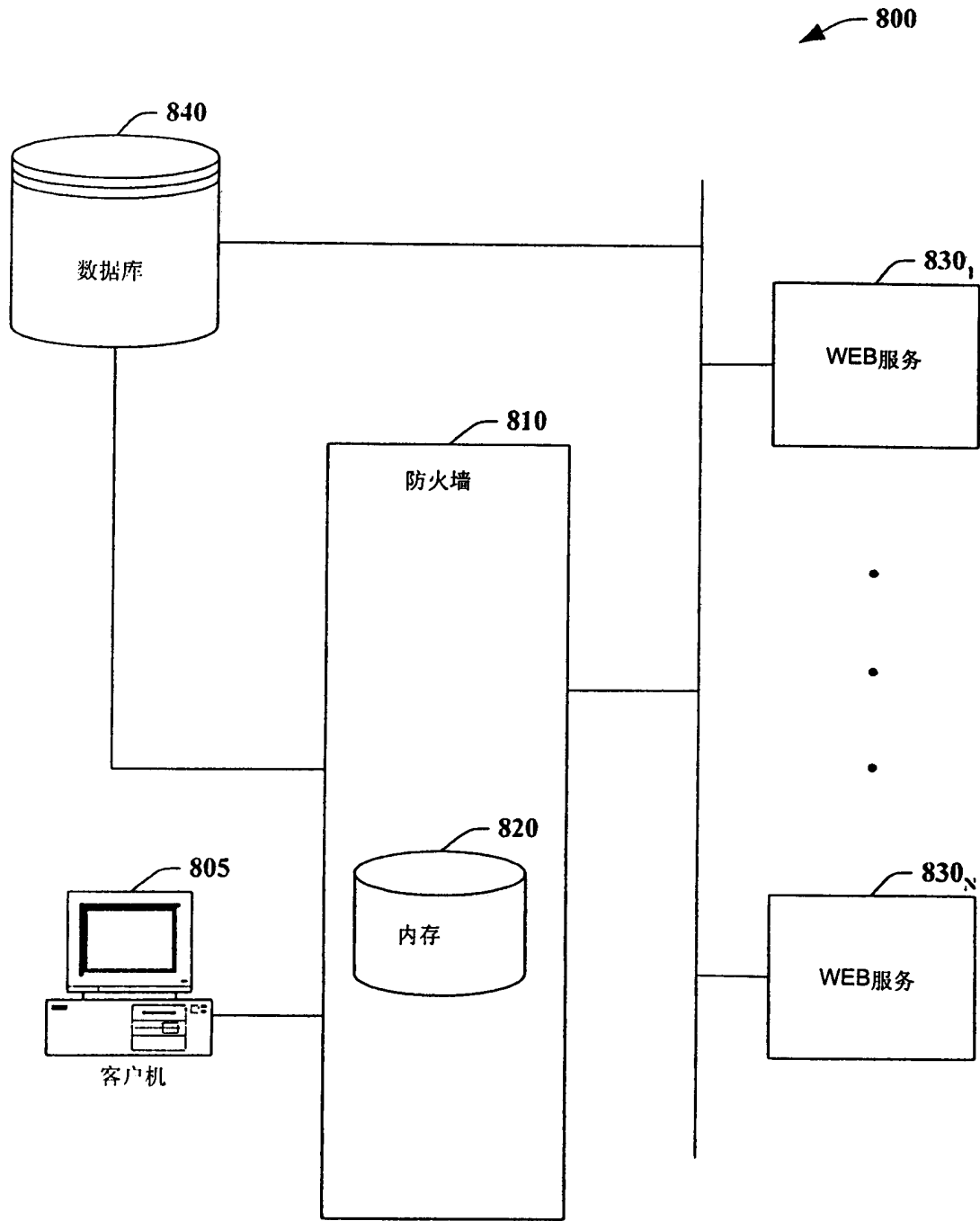


图 8

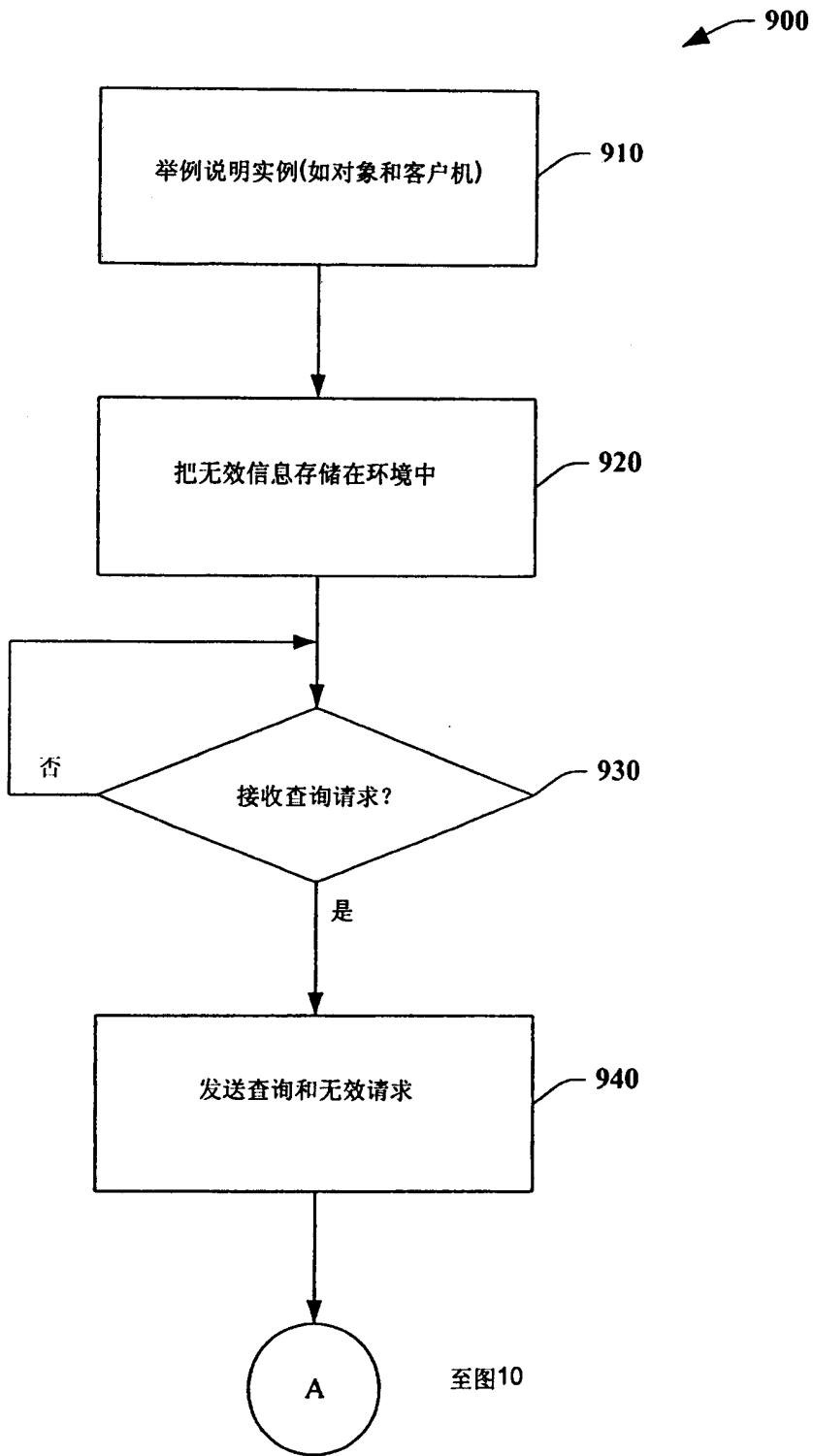


图 9

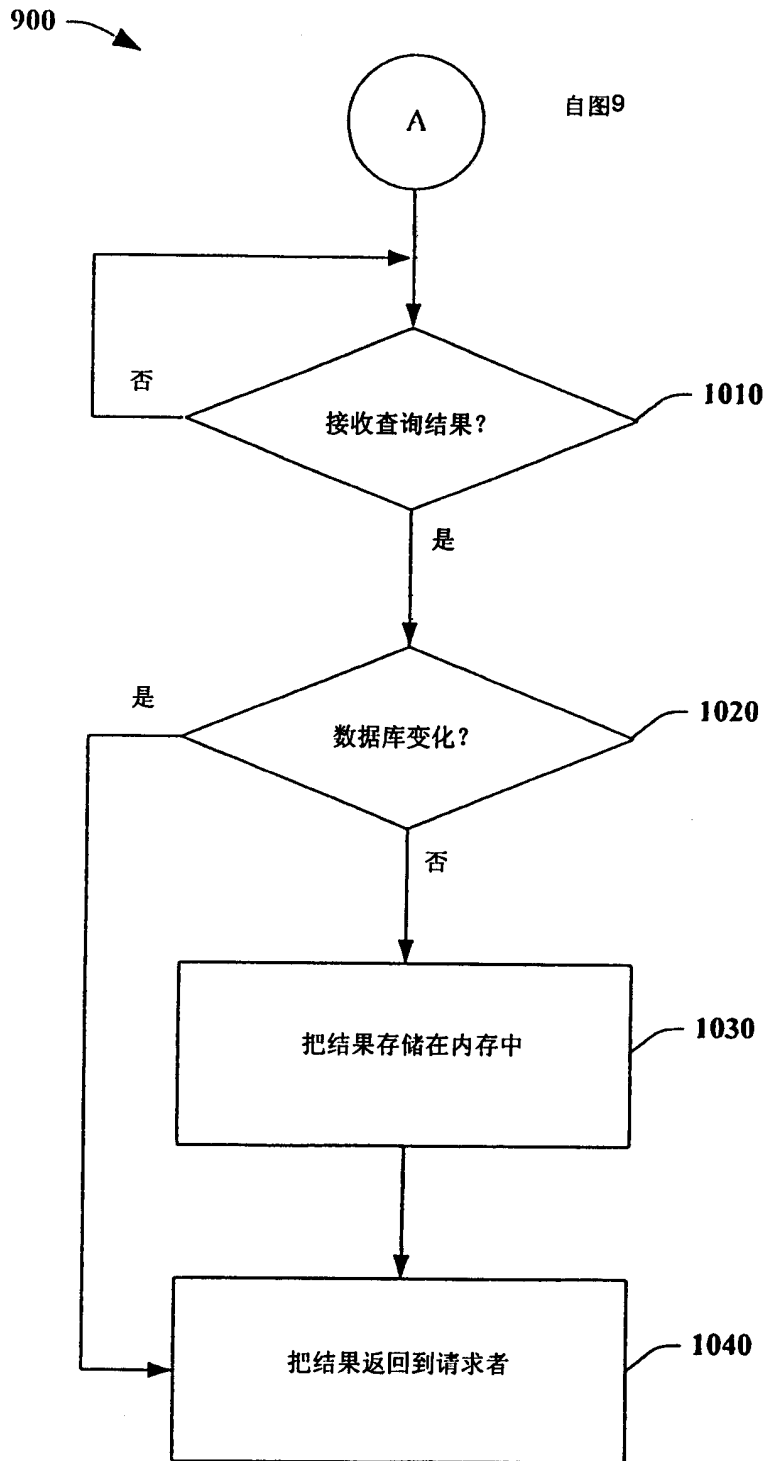


图 10

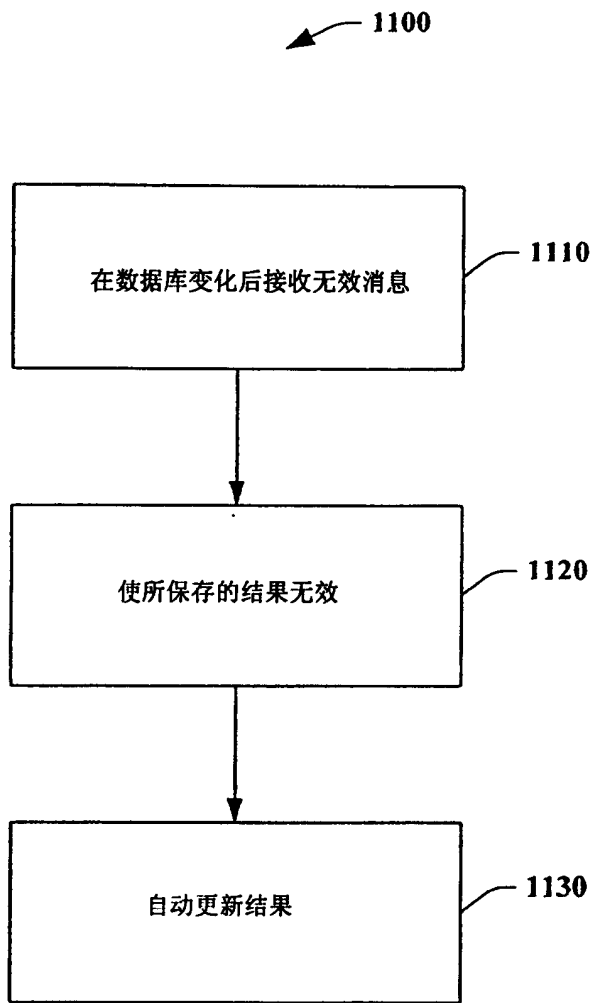


图 11

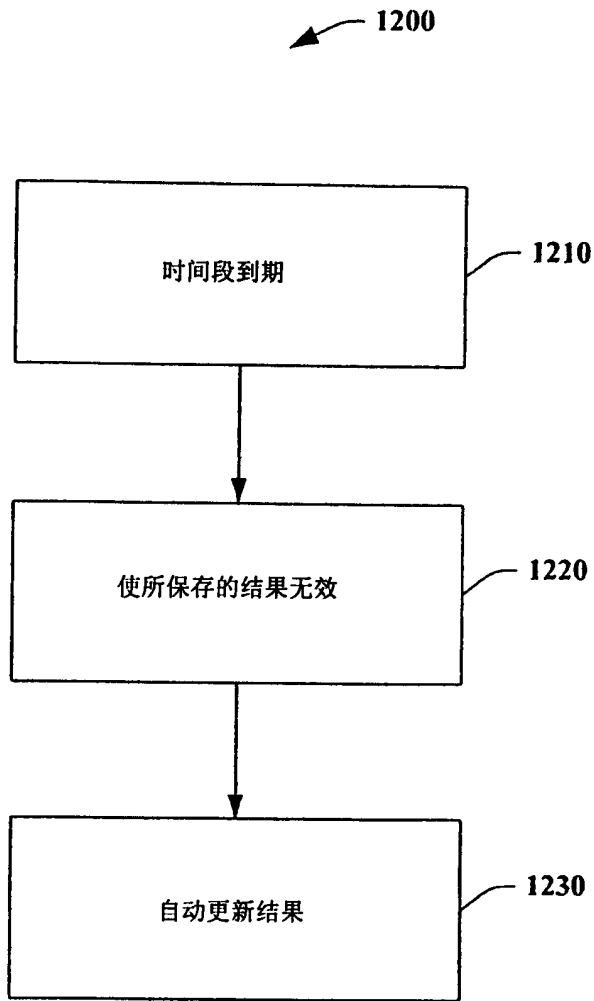


图 12

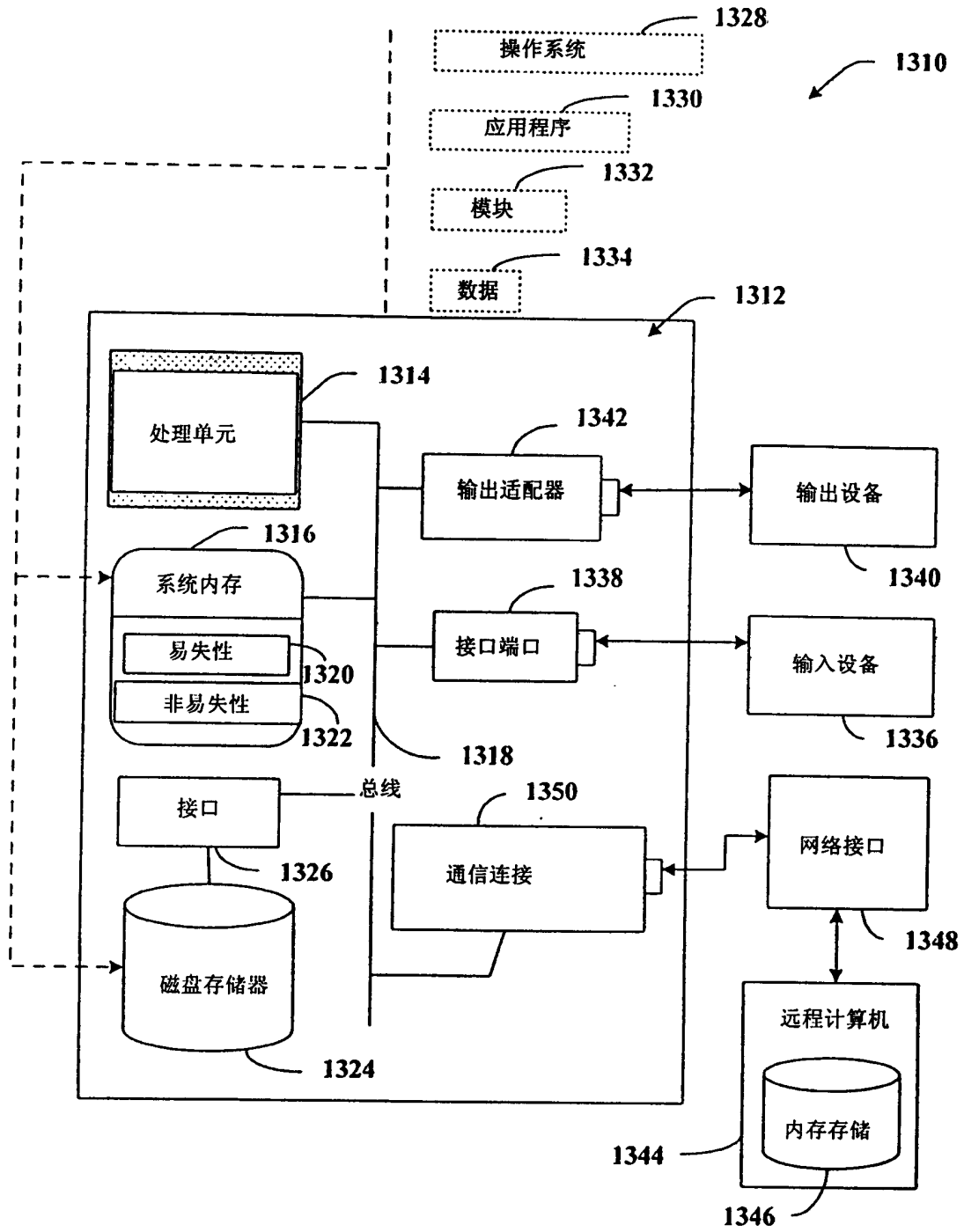


图 13