

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06F 17/30 (2006.01)

G06F 12/00 (2006.01)

H04L 12/00 (2006.01)



[12] 发明专利说明书

专利号 ZL 200410055078.X

[45] 授权公告日 2009 年 2 月 11 日

[11] 授权公告号 CN 100461162C

[22] 申请日 2004.6.25

[21] 申请号 200410055078.X

[30] 优先权

[32] 2003.6.25 [33] US [31] 10/603,532

[73] 专利权人 微软公司

地址 美国华盛顿州

[72] 发明人 M·J·皮佐 R·M·海华德

P·Y·-K·吴 S·D·古斯里

A·W·史密斯

[56] 参考文献

CN 1366244 A 2002.8.28

EP 1178402 A1 2002.2.6

US 6026413 A 2000.2.15

审查员 刘长勇

[74] 专利代理机构 上海专利商标事务所有限公司

代理人 陈 斌

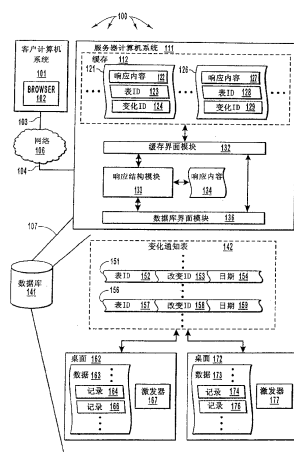
权利要求书 4 页 说明书 27 页 附图 4 页

[54] 发明名称

用于寄存和检索可以用来使缓存记录无效的数据库表格变化信息

[57] 摘要

一服务器提供可以包括来自数据库中数据表的内容的网络响应。该服务器提供一可以存储内容(包括来自数据表的内容)的缓存(例如,在系统存储器)以提高随后提供相同内容的效率以满足客户端网络请求。该服务器监视数据表的变化并且,当在一特定的数据表中出现一变化时,使依赖于一特定数据表的缓存记录无效。进而,响应于一客户端网络请求的网络响应,该服务器给至少一部分所构建的网络响应分配数据库缓存依赖性,该部分所构建的网络响应是基于在构建该网络响应期间所执行的命令的。该至少一部分所构建的网络响应随后被缓存在服务器的一缓存位置。



1. 在一访问具有一个或多个数据表的计算机系统中，一种用于当其中一个数据表的内容被替换时，配置数据库以提供一表变化通知的方法，该方法包括：

选择要被用于监视内容变化的数据表的步骤；

把一条相应该选定数据表的记录插入到一变化通知表的步骤，该相应记录包括选定数据表的版本信息；

给选定数据表分配触发器的步骤，该触发器导致当在选定数据表中的内容被替换时所选定数据表的版本信息在一变化通知表中被更新；

在响应于选定数据表中被替换的内容的变化通知表中更新版本信息的步骤；

以及

返回该更新的版本信息至一请求中的服务器计算机系统的步骤，从而使被更新的版本信息可以被用来决定在服务器计算机系统的缓存记录中的内容的有效性。

2. 如权利要求1所述的方法，其中选定一要被用于监视内容变化的数据表的步骤包括接收用户输入来选择要被用于监视内容变化的一数据表的步骤。

3. 如权利要求1所述的方法，其中选定一要被用于监视内容变化的数据表的步骤包括计算机系统响应于一接收的网络请求自动选择一数据表的步骤。

4. 如权利要求1所述的方法，其中把一条相应该选定数据表的记录插入到一变化通知表的步骤包括响应于用户输入插入记录的步骤。

5. 如权利要求1所述的方法，其中把一条相应该选定数据表的记录插入到一变化通知表的步骤包括计算机系统响应于一个网络请求自动插入该记录的步骤。

。

6. 如权利要求1所述的方法，其中，所述数据库是SQL数据库。

7. 如权利要求1所述的方法，其中给选定数据表分配触发器的步骤包括接收构建一要被分配给所选定数据表的触发器的用户输入的步骤。

8. 如权利要求1所述的方法，其中给选定数据表分配触发器的步骤包括计算机系统响应于接收一包含在选定数据表中的内容的网络请求自动分配一触发器给选定数据表的步骤。

9. 如权利要求1所述的方法，其中给选定数据表分配触发器的步骤包括分配

一个触发器的步骤，当触发器响应于所选定数据表中的内容被替换而执行时，将在该变化通知表中更新一相应的变化ID。

10. 如权利要求1所述的方法，其中响应于选定数据表的内容被替换在一变化通知表中更新版本信息的步骤包括一执行该触发器的步骤。

11. 如权利要求1所述的方法，其中返回该更新的版本信息至一请求中的服务器计算机系统的步骤包括一个返回所更新的版本信息至一请求中的构建包括来自选定数据表内容的网络响应的计算机系统的步骤。

12. 在一提供网络响应至请求中的客户端计算机系统的服务器计算机系统中的一种用于使一部分缓存的内容无效的方法，该网络响应潜在地包括来自在数据库中的一个或多个数据表的内容，该服务器计算机系统包括一个缓存定位器，该缓存定位器存储预先从一数据表接收所述一部分缓存的内容和相应于所述缓存的内容的缓存的版本信息，该方法包括下列内容：

- 一查询一变化通知表的相应于该一个或多个数据表的版本信息的步骤；

- 一接收自从该服务器计算机系统最后查询变化通知表以来已经改变的至少一个数据表的列表的步骤，该至少一个数据表的列表包括该至少一个数据表的当前版本信息；

- 一决定所缓存的内容依赖于至少一个数据表的步骤；

- 一比较当前版本信息和所缓存的版本信息的步骤；以及

- 一基于比较结果从缓存定位器移除该部分缓存的内容的步骤。

13. 如权利要求12所述的方法，其中查询一变化通知表的相应于该一个或多个数据表的版本信息的步骤包括一发送一查询至该数据库的步骤。

14. 如权利要求13所述的方法，其中发送一查询至该数据库的步骤包括一发送一指定所缓存的版本信息和仅选择那些不再匹配所缓存的版本信息的表和版本信息的步骤。

15. 如权利要求14所述的方法，进一步包括：

- 一接收一空结果作为一个没有表的版本信息被替换的结果的步骤。

16. 如权利要求14所述的方法，其中发送一查询至该数据库的步骤包括发送一在数据库中的阻塞直到使所请求的表版本信息产生变化的查询的步骤。

17. 如权利要求16所述的方法，其中发送一在数据库中阻塞直到使所请求的表版本信息产生变化的查询的步骤包括一发送一阻塞一在服务器计算机系统的

相应线程的查询直到所请求的表版本信息的一个变化被检测到的步骤。

18. 如权利要求16所述的方法, 其中发送一在数据库中阻塞直到使所请求的表版本信息产生变化的查询的步骤包括一在服务器计算机系统注册一回叫信号的步骤, 该回叫信号被配置为当所请求的表版本信息的变化被检测到时被呼叫。

19. 如权利要求13所述的方法, 其中发送一查询至该数据库的步骤包括一发送一请求所有被服务器计算机系统监视的表的版本信息的查询的步骤。

20. 如权利要求19所述的方法, 进一步包括:

一接收所有被监视表的所请求的版本信息的步骤; 和

一服务器计算机系统比较所接收的版本信息和服务器计算机系统所维持的版本信息的步骤。

21. 如权利要求12所述的方法, 其中查询一变化通知表的相应于一个或多个数据表的版本信息的步骤包括一查询一变化通知表的相应于一个或多个数据表的变化ID的步骤。

22. 如权利要求12所述的方法, 其中该接收自从该服务器计算机系统最后查询变化通知表以来已经改变的至少一个数据表的列表的步骤包括一接收一个或多个版本信息数据结构的一个列表的步骤。

23. 如权利要求12所述的方法, 其中决定所缓存的内容依赖于至少一个数据表的步骤包括一决定一缓存记录依赖于—主码记录的步骤, 该主码记录依赖于—数据表从而使该主码记录的无效会自动使该缓存记录无效。

24. 如权利要求12所述的方法, 其中比较当前版本信息和所缓存的版本信息的步骤包括一比较—所接收的变化ID和—所缓存的变化ID的步骤。

25. 如权利要求12所述的方法, 其中基于比较结果从缓存定位器移除该部分所缓存的内容的步骤包括一响应于—主码记录的无效而自动使—依赖缓存记录无效的步骤。

26. 如权利要求1所述的方法, 其特征在于, 一数据表的版本信息由一数据结构表示, 所述数据结构包括:

一表示可以用来识别一数据表的表ID的表ID域; 和

一表示由在表ID域中的表ID识别的数据表的变化ID的变化ID域, 该变化ID被用来决定一缓存记录是否要被无效。

27. 如权利要求26所述的方法，所述数据结构进一步包括：

—表示数据库开始寻找数据表的版本的日期的日期域。

28. 如权利要求26所述的方法，所述数据结构进一步包括：

—表示依赖于由表ID域中的表ID所识别的数据表的页面内容的页面内容域

。

用于寄存和检索可以用来使缓存记录 无效的数据库表格变化信息

技术领域

本发明涉及管理缓存内容。尤其是，本发明涉及用于寄存和检索可以用于无效缓存记录的数据库表格变化信息。

背景技术

计算机系统和相关技术影响社会的许多方面。确实，处理信息的计算机系统性能已经改变了我们生活和工作的方式。计算机系统现在通常执行一系列优先于手动执行的计算机系统出现的主要的任务（例如，文字处理，制表，和数据库管理）。近来，通过计算机系统可以电子通信并共享数据的有线或无线计算机网络计算机系统已经被彼此耦合了。结果，在计算机中处理的许多任务（例如，语音通信，访问电子邮件，电子会议，网络浏览）包括通过有线和/或无线计算机网络与一个或多个其它计算机系统电子通信。

特别地，万维网（“WWW”）成为交流信息的重要机制。一般来说，在WWW上的内容是在一客户端/服务器模式中被访问。在一个客户端计算机系统上的一个“网络浏览器”发送一个访问通过在服务器计算机系统上的“网络服务器”被维持的内容的请求。到适当的时候，网络服务器持续地提供所请求的内容给该网络浏览器。

一种使内容对于网络浏览器来说是可用的方式是为了给网络开发商创建一个包括若干个硬盘-编码的计算机可读指令的静态网页，例如，举例来说，在超文本标记语言中（“HTML”）。然后该静态网页被存储在一个对网络服务器可访问的位置（例如，在服务器的计算机系统，在一网络可连接的数据库中，等等）。响应于来自网络浏览器的一个请求，网络服务器可以将该静态网页传送至网络浏览器（例如，作为用户输入的结果导致网络浏览器访问一个相应于该静态网页的统一资源定位器（“URL”））。由于静态网页的接收，网络浏览器可以在客户端计算机系统上表示该静态网页的内容（例如，文本，图形，等等）。

为了保存网络资源，网络浏览器也可以在客户端计算机系统（例如，在系统存储器或在一大容量存储设备中）上缓存一个静态网页的副本。当静态网页被不断地请求时，该网络浏览器会在初始化一个新网络请求到网络服务器之前尝试从缓存中定位该静态网页。如果该静态网页被定位在缓存中，网络浏览器会检索被缓存的静态网页而不是初始化一个网络请求。由于本地存储的数据可以被更有效地访问，因此在客户端计算机系统上缓存静态网页会减少与静态网页显示的内容相关的潜伏性。

对静态网页的频繁的请求也可能也会被缓存在一个服务器计算机系统中（一般涉及诸如输出缓存）以减少潜伏性。例如，一个频繁请求的静态网页可以被缓存在服务器计算机系统的系统存储器中以至于不必每次当该静态网页被请求时都必须从大容量的存储设备中检索该静态网页。因为它们的静态本质，静态网页可以被缓存（在一客户端或服务器计算机系统中）很长的时间周期，潜在性不确定。

然而，静态网页的使用主要是阻止实时或自定义数据在网络浏览器的存在。一般来说，在网络浏览器中需要存在的 是动态数据（例如，日常比例图片，股票现价，等等）。因此，在动态网页中的包括动态数据的要在一网络浏览器中显示的机构已经被开发出来了。一个用于包含在动态网页中动态数据的机构是作为计算机可执行指令（一般称为“脚本”）的模块件使用的。

脚本可以使用脚本语言来开发，例如，像这样的，JavaScript和VBScript，和在一页面文件（例如，一活动服务器页面（“ASP”）文件）中嵌入HTML和扩展标记语言（“XML”）指令。当接收到一个动态网页的请求时，服务器计算机系统将请求重发到一个处理相应页面文件的处理模块中（例如，一ASP模块）。处理模块执行包括在页面文件的任何脚本，并且生成适当的HTML指令返回给相应的请求。脚本可以包括用于从动态位置访问数据的计算机可执行指令，例如，像这样的，从一个包括当前比例图片或股票报价的数据库。由于脚本主要在接收请求时被执行，因此脚本可以被用来生成包含实时数据的动态网页。

类似于静态网页，动态网页也可以被缓存以提供对动态网页的更为有效的访问。然而，包括在动态网页中的动态内容（例如，价格报价）对时间限制是特别有效的（例如，直到在动态内容位置处被检索到的值已经被改变）。一般

来说，动态内容（例如，价格报价）只会在一个很短的时间内有效，例如，只有几秒钟。当动态网页内容的值改变时，缓存的动态内容就变得无效了。例如，当股票价格从\$20.00变为\$21.00，缓存的\$20.00的值就是无效的（或者甚至是正确的）。当无效的动态内容仍然在缓存中，对于网络服务器来说就有给网络浏览器提供错误的动态内容的潜在性。例如，当一个无效的股票价格仍然保存在缓存中时，网络服务器可以从缓存中检索到该无效的股票价格而不是重新执行一个脚本以从数据库中检索一更新过的股票价格。因此，已经开发出了尝试确保缓存动态内容有效性的缓存管理机制。

一些缓存管理机制在指定的时间间隔中会自动从缓存中移除动态内容，例如，每十分钟。然而，如果动态内容的值在指定的时间间隔中改变那么缓存中仍然存在一种无效的动态内容存在于缓存中的潜在性。例如，如果指定时间间隔是五分钟并且动态内容值在被缓存后一分钟更新，动态内容值是无效的但依然会缓存至少四分钟。在这四分钟里，网络服务器可能会不正确地检索缓存的动态内容值而不是执行一个脚本去检索该已经更新的动态内容值。

而且，在指定时间间隔移除内容会导致仍然有效的值从缓存中移除。例如，如果指定的时间间隔是十分钟并且动态内容值在被缓存后在十分钟内仍然有效，那么该动态内容值依然被从缓存中移除。因此，更有效检索相应后续请求的动态内容值的潜在性被阻止。

其他缓存管理机制监视存储在服务器计算机系统上的文件（例如，XML文件）并且当被监视的文件改变时使从监视的文件得到的缓存动态内容值无效。而其他缓存管理机制监视缓存中的记录并在其他缓存记录无效时使一个缓存记录无效。因此，一些缓存管理机制导致缓存记录的有效性依赖于时间，一个相应的文件，或其他相应的缓存记录。

然而，现存的缓存管理机制（包括那些使用的时间，文件，和缓存依赖性）对于使从计算机系统外部的地址中检索到的缓存的动态数据值无效还不是很合适。例如，网络浏览器可以请求一个包括来自数据库中若干个不同的表格的动态内容值的网页。因此，一个具有合适脚本的页面文件可以在服务器计算机系统上被处理从而检索动态内容值并且生成适当的HTML指令。

然而，从数据库中检索数据要消耗网络和数据库资源。如果在检索的数据上还要执行计算，那么还要额外消耗服务器计算机系统资源。而且，可能会这

样，服务器计算机系统同时询问数据库要求被包括在动态网页内的动态内容的大量不同部分。也有可能大量服务器计算机系统每个都从相同的数据库中访问动态内容。

因此，一个单个的数据库可能会同时给大量的服务器计算机系统提供动态内容，其中每个服务器计算机系统同时询问数据库要求动态内容的不同部分。结果，单个服务器成为在大量服务器计算机系统共享资源并且因此潜在限制也缩小了。也就是，更多服务器计算机系统的额外增加会不必要地引起运行的增加，由于每个服务器计算机系统至少受从单个数据库中检索内容所需时间的限制。而且，当从缓存（例如在一指定间隔）中预先完全移除了内容时服务器计算机系统会被要求再次询问数据库，甚至在缓存中的动态内容依然有效。在多服务器环境中，缓存中内容的预先完全移除会导致共享数据库中的资源的不必要的负担，因此，而且是很重要的，要限制可量测性。因此，依赖于配制缓存中记录的机制和使基于数据库中表格的变化的缓存记录无效是有好处的。

发明内容

背景技术中的前述问题已经被本发明的原理所克服，其中导出了系统，方法，计算机程序产品，和用于注册检索可以用来使缓存记录无效的数据库表格变化信息的数据结构。具有浏览器的客户端计算机系统对可生成并提供网络响应的（例如，网页和/或网络服务结果）服务器计算机系统是网络可连接的。服务器计算机系统至少访问从数据库中数据表格响应的部分内容。服务器计算机系统保存可以存储已提供内容的缓存以提高后面提供相同内容至安全客户端网络请求的效率。

服务器计算机系统选择一个包含在用来监视数据变化的数据库中（例如SQL数据库）的数据表格（例如，响应于用户输入）。相应于选定数据表格的一条记录被插入到一变化通知表中（也包含在数据库中）。相应的记录包括版本信息，例如，像这样的，一个对于被选定数据表的，变化ID。一个触发器，其导致选定数据表的版本信息当在选定数据表中的内容被替换时在变化通知表中被更新。

响应于在选定数据表中的变更内容（例如，插入，删除，或更新一条记录），在变化通知表中的相应版本信息被更新。有时或者响应于检索缓存内容的请求，服务器计算机系统查询变化通知表相应于数据库中一或多个表的版本信息。一个

查询可以被配置为请求所有显示在变化信息表中的监视数据表的版本信息，请求具有突出缓存依赖性的被监视的数据表的版本信息，或者请求已变化（例如，通过包括在查询中当前版本信息）的被监视数据表的版本信息。响应于一个查询，更新的版本信息从数据库中返回到服务器计算机系统。基于查询的结构，数据库可以返回，例如用于所有被监视的数据表的版本信息，具有突出缓存依赖性的被监视的数据表的版本信息，或者请求已变化的被监视数据表的版本信息。

可替代地，服务器计算机系统可以发布一个查询到变化表并请求将所发布的查询阻塞在该服务器计算机系统直到所请求的表的版本信息被更新。例如，服务器计算机系统可以发布阻塞在服务器计算机系统上的查询直到被当前的版本信息比一指定值大。发布的阻塞查询可以被同步执行，以至于在服务器计算机系统上的一个相应的线程不被阻塞并且当结果可用时调用一个复查信号。

服务器计算机系统决定了缓存的内容是依赖于一个数据表的（例如，作为一个前面指定的数据库缓存依赖性的一个结果）。服务器计算机系统比较当前的版本信息（在接收表中的）和所缓存的版本信息（例如，与一部分缓存内容缓存在一起的版本信息）。这可以包括比较一接收的变化ID和一缓存的变化ID。当适当时，例如，当在数据表中的内容在基于数据表的一部分内容被缓存后改变时，部分缓存的内容就被从缓存中移除。因此，依赖于受监视的数据表的缓存的内容可以被适当地无效并从缓存中被移除。适当的无效和移除缓存中的内容在一网络响应中会减少包括无效缓存内容的服务器计算机系统的可能性。而且，移除有效数据（例如，可以表示为与时间相依赖的数据）的机会也因此而减少了因此潜在地提高了网络响应发生的效率。

在一些实施例中，缓存的内容就是以用于生成网络响应的对象和结果的形式，例如表格式数据，对象，数据采集，或局部响应。当这些结果可以被用在超过一个的网络响应的发生中时缓存这些中间结果是合适的，其中每个网络响应包含额外的差异信息（例如，其他相关信息的定制或包含）。在其他一些实施例中，一个完整的网络响应被缓存。当该完整的网络响应可以通过多个网络请求共享时缓存该完整的网络响应是合适的。当缓存一个完整的网络响应时，该网络响应可以在服务器计算机系统中被推迟并以低级别缓存，例如，一个内核模式的响应缓存。因此，当该网络响应无效时，它就被从低级别缓存中移除。

可以是这样的，服务器计算机系统接收一个客户端关于一网络响应的包括

被包含在数据库中的所请求内容的网络请求。数据表可以是一种对于其他可以在数据表中替代内容的应用来说是可访问的数据表，包括所请求的内容。服务器计算机系统决定该被包含在网络响应中的至少一部分内容当前并未缓存在服务器计算机系统中。服务器计算机系统从数据表中访问这些所请求的内容并且构建该包括所请求内容的网络响应。

服务器计算机系统为至少一部分所构建的基于在网络响应构建期间执行的命令的网络响应分配数据库缓存的依赖性。分配一个数据库缓存的依赖性可以包括分配依赖于数据表的网络响应或分配基于从依赖于数据表的数据表中接收的内容。数据库缓存的依赖性可以通过指示的使用或包含页面文件（例如，在活动服务器页面中（“ASP”））的程序代码来分配，该页面文件是在服务器计算机系统被处理以构建的该网络响应。至少一部分被构建的网络响应被缓存在服务器计算机系统中。因此，响应于后续的被构建的网络响应的网络请求，服务器计算机系统可以检索该已缓存的被构建的网络响应的至少一部分。检索缓存内容（潜在地为一完整的网络响应）不仅保存了服务器计算机系统上的资源（因为部分页面文件未被执行）以及网络和数据库资源（因为数据库没有被查询从而数据未被在网络中传送）。因此，通过减少处理过程和响应于网络请求的资源要求减少了潜伏性的存在，而通过减少同时访问共享资源例如数据库量测性则提高了。

本发明的更多的特性和优点将在下述说明书中披露，并且可从说明书中部分通晓，或者可通过本发明的实例中导出。本发明的特性和优点可以通过仪器的方式或者所附的权利要求书中指出的特定的结合被实现或包含。本发明的这些和其他一些特点从下面的描述和权利要求书中将变得更为完整清楚，或者可以从本发明后面阐述的实施例中导出。

附图说明

为了描述上述方式和本发明的包含的其他优点以及特性，上面简明描述的本发明的更为详细的描述将通过参考包括所附附图的指定实施例来演示。应该领会到那些附图的描述仅仅是用来描述某些典型的实施例的而不是来限定其范围的，本发明将通过使用所附的附图来描述并解释额外的特性及细节：

图1示出了依据本发明原理的一个配置了依赖于其上的缓存记录的数据表中的变化无效的设备的网络结构的例子。

图2示出了依据本发明原理的一种用于配置一个数据库以提供表格变化通知的方法的一个流程图的例子。

图3示出了依据本发明原理的一种用于使缓存内容部分无效的方法的一个流程图的例子。

图4示出了依据本发明原理的一种用于访问被客户端计算机系统请求显示的方法的一个流程图的例子。

图5示出了一种用于本发明原理适合的操作环境。

具体实施方式

本发明涉及方法，系统，计算机程序产品，和用于注册和检索可以用来使缓存记录无效的数据库表格变化信息的数据结构。具有浏览器的客户端计算机系统对于可以生成和提供网络响应给客户端计算机系统（例如，网页和/或网络服务的结果）的服务器计算机系统是可网络连接的。服务器计算机系统至少从数据库中的数据表中访问网络响应的部分内容。服务器计算机系统包括一个可以存储所提供内容（包括来自数据表的内容）的缓存（例如，在系统存储区中）以提高所后提供相同的内容至安全客户端网络请求的效率。

服务器计算机系统（例如，响应于用户输入）配置了数据库中的一个数据表以在当数据表中的内容被替换（例如，插入，删除，更新）时提供一个关于更新的版本信息（例如，改变ID）的变化通知表。服务器计算机系统不时地查询该变化通知表或者响应于检索所缓存内容以请求相应数据表的已更新的版本信息。在响应中，数据库提供相应于该查询的数据表的已更新的版本信息。服务器计算机系统可以使用已更新的版本信息以使缓存记录（例如，基于预先分配的数据库缓存依赖性）无效。

在一些实施例中，服务器计算机系统接收一用于包括包含在数据表中的所请求内容的网络响应的客户端网络请求。服务器计算机系统决定被包含在网络响应中的内容的至少一部分当前没有缓存在服务器计算机系统中。服务器计算机系统从数据表中访问所请求的内容并且构建该网络响应以包括所请求的内容。服务器计算机系统给基于在构建该网络响应期间所执行的命令所构建的网络响应的至少一部分分配了一数据库缓存依赖性。至少一部分所构建的网络响应被缓存在服务器计算机系统中。

本发明的一些实施例可以包含特定的用途或一般用途计算机，其包括各种

各样计算机硬件和软件的计算机，下面将具体阐述。特别地，在本发明的范围内的实施例包括用于携带或具有计算机可执行指令或存储于其上的数据结构的计算机可读媒体。这种计算机可读媒体可以是任何可以通过特定应用计算机或一般应用计算机访问的可用媒体。举个例子，但并不限于此，这种计算机可读媒体可以包含RAM，ROM，EEPROM，CD-ROM或其他物理存储媒体，例如光学磁盘存储，磁盘存储或其他磁性存储设备，或任何其他可以以计算机可执行指令或数据结构的形式用来记载或存储所需程序代码的方法的介质并且该介质可以被特定应用计算机和一般应用计算机访问的。当信息通过网络或其他与计算机的通信连接（硬件线路，无线，或硬件线路和无线结合的连接）被传送或提供时，该计算机可以适当地把该连接视为一计算机可读介质。因此，任何这种连接可以称为一计算机可读介质。上述方式的结合也可以包含在计算机可读介质范围内。计算机可执行指令包括，例如，会导致一般应用计算机，特殊应用计算机，或特殊应用处理设备，例如一GPU去进行一特定功能或一组功能的指令和数据。

在该描述中和在后面的权利要求中，“计算机系统”被定义为一个或多个软件模块，一或多个硬件模块，或它们的结合，它们结合在一起以处理指令或电子数据。例如，计算机系统的定义包括个人计算机的硬件元件，软件模块也一样，例如个人计算机的操作系统。模块的物理设计是不重要的，计算机系统包括一个或多个通过网络耦合的计算机。同样地，计算机系统也可以包括一个单个的其内部模块（例如存储器和处理器）一起工作处理指令和电子数据的物理设备（例如一移动电话或个人数字助理“PDA”）。

在该描述中和在后面的权利要求中，一“网络响应”被定义为一人类可读的页面，例如，像这样的，一网页，和/或来自网络服务的结果。网页可以从网络检索，例如，像这样的，通过访问相应的统一资源定位器（“URL”）的万维网（“WWW”）。网页可以包括个一或多个可显示在网络浏览器上的超文本标记语言（“HTML”）指令，例如，像这样的，浏览器102。网络服务提供用于不同应用和不同资源之间进行通信（用户交互作用的潜在依赖性）的机制而不必开发用户指令。网络服务可以集成一些不同的技术，例如，像这样的，用于标记数据的扩展标记语言（“XML”），用于传送数据的简单对象访问协议（“SOAP”），用于说明可用服务的网络服务说明语言（“WSDL”），和

用于列出可用服务的通用说明，发现和综合语言（“UDDI”）。网络服务可以共享事务逻辑，数据，和通过网络的程序界面进行处理。

在该描述中和在后面的权利要求中，“网络请求”被定义为对于“网络响应”的请求。也可以是这样，用户在用户界面输入适当的命令以初始化一网络请求。例如，用户在客户端计算机系统101的浏览器102可以输入适当的命令以访问一相应网页的URL。另一方面，也可以通过应用来初始化一网络请求。例如，在客户端计算机系统101的网络服务应用可以初始化一网络请求至服务器计算机系统111的网络服务应用。

本领域的技术人员将会明白本发明可以应用在许多种计算机系统配置之中，包括，个人计算机，膝上电脑，多处理器系统，微机，大型计算机，及其类似。本发明也可以使用在包括本地和远程计算机系统的分布式系统环境中使用，它们通过网络相连（或者通过硬件线路相连，无线相连，或者通过硬件线路和无线的结合相连）一起执行任务，在分布式系统环境中，程序模块和相关数据结构可以位于本地或远程存储器存储设备中。

现在来看图1，图1示出了依据本发明原理的一个配置了依赖于其上的缓存记录和基于数据表中的变化无效的设备的网络结构100的例子。网络结构100中的矩形元件（例如，浏览器102，缓存界面模块132，响应结构模块133，数据库界面模块136，触发器167，和触发器177）表示促进本发明原理的可执行模块。卷形元件（缓存121和126，响应内容134，记录151和156，以及数据163和164）表示由可执行模块处理的数据。因此，网络结构100中描述的可执行模块和卷形元件根据本发明的原理相互协作。

在网络结构100中，客户端计算机系统101和服务器计算机系统111通过相应连接103和104分别与网络106相连的。网络106可以是局域网（“LAN”），广域网（“WAN”），或者甚至是因特网。客户端计算机系统101包括浏览器102，它可以发送网页的网络请求（例如，超文本传送协议（“HTTP”）请求）并接收响应于一网络请求所返回计算机系统101的网页。浏览器102也可以在客户端计算机系统101上显示所接收的网页内容（例如，文本，图形，视频，和音频）。客户端计算机系统101也可以包括与在服务器计算机系统111上的相应的网络服务应用通信的网络服务应用。

服务器计算机系统111包括可以构建响应客户端网络请求的网络响应的响

应结构模块133。响应结构模块133可以处理页面文件，例如，像这样的，动态服务器页面（“ASP”）文件，以构建网页。页面文件可以包括多种不同语言形成的指令和/或命令，例如，像这样的，超文本标记语言（“HTML”），扩展标记语言（“XML”），JavaScript，和VBScript。响应结构模块133可以处理页面文件中的指令和/或命令以生成网络响应的内容。响应结构模块133可以被配置以生成网页，例如，从浏览器响应于一网络请求，并配置以生成网络服务结果，例如，从一网络服务响应一网络请求。

服务器计算机系统111通过连接107与数据库141（例如，一SQL数据库）相连。数据库界面模块136可以通过连接107传送数据库命令至数据库141。数据库命令可以被用来在数据库141中处理内容（例如，添加，删除，或改变行或记录）或从其中检索内容。数据库界面模块136可以接收用户输入（例如，从键盘或鼠标），其可以被翻译成数据库命令并被传送到数据库141。可替代地，数据库界面模块136可以从在一网络响应中包含内容的响应结构模块133接收输入。在响应中，数据库界面模块136可以从数据库141中检索内容并提供该已检索的内容至响应结构模块133。响应结构模块133可以将响应内容134中的静态内容与所检索的内容相结合以构建一个适当的网络响应。

数据库141可以存储包含在数据库141中的数据表的版本信息。数据库界面模块136可以检索数据表的版本信息并提供所检索的版本信息至缓存界面模块132中。缓存界面模块132可以利用所检索的版本信息来决定何时使缓存112中的记录无效（并在之后删除）。例如，基于所检索的版本信息缓存界面模块132可以决定何时使缓存记录121或缓存记录126无效。缓存112可以是一个具有降低潜在性的存储位置，例如，像这样的，服务器计算机系统111的系统存储器。

版本信息可以被存储在表示被包含在数据库141中的数据表的版本的版本信息数据结构中。版本信息数据结构的一个域可以是表示鉴定数据库141中数据表的表ID的表ID域。一个表ID可以是一个值，例如，可以唯一标识一个数据表的，包括数字和/或字母的字符。例如，表ID“销售表3”可以用来识别表162。

版本信息数据结构的另一个域可以是一个表示由一相应表ID识别的数据表的一变化ID的变化ID域。变化ID域可以是一个值，例如，可以识别识别数据表的当前版本的，包括数字的字符。当数据表中的内容被改变时，相应的变化ID可以被更新（例如，通过增加该变化ID）。例如，当内容被插入到表172中时，

变化ID158可以被递增。版本信息数据结构的另一个域可以是表示数据库141何时开始寻找由相应表ID识别识别的数据表的版本的日期的日期域。例如，日期154可以说明何时数据库141开始寻找表162的版本信息。因此，版本信息数据结构可以是下面的形式：

版本信息

```
{  
  表ID域  
  变化ID域  
  日期域  
}
```

一个或多个版本信息数据结构可以被存储为包含在数据库141中的变化通知表的记录。例如，变化通知表142包括记录151和156。而且，在变化通知表142中每一组包括三个竖向的省略号表示变化通知表142可以包括其他记录。相应于数据表的在变化通知表142中的每个记录都被配置以寻找版本信息。例如，记录151和156可以分别与表162和172对应。因此，表ID152可以存储一个用以识别识别表162的值，变化ID153可以存储一个说明表162当前版本信息的值，日期154可以存储说明数据库141何时开始寻找表162的版本信息的值。同样地，表ID157可以存储一个用来识别识别表172的值，变化ID158可以存储一个说明表172当前版本的值，并且日期159可以存储一个说明数据库141何时开始寻找表172的版本信息的值。

一个数据表可以被配置为通过使用相应的在一变化通知表中更新一适当变化ID的数据库触发器（例如，一部分计算机可执行指令）来寻找版本信息。当数据表中的内容被替换时，数据库触发器被执行并且适当的变化ID被更新。例如，当在记录164中的数据被替换，触发器167可以被执行以更新变化ID153的值。类似地，当记录176被插入到数据173中时，触发器177可以被执行以更新变化ID158的值。数据库141（或服务器计算机系统111的备用系统）的一个管理员或用户可以输入导致与数据表相关的适当的触发器上的用户输入。

表162和表172是包含在数据库141中的数据表。表162和172中的每一个都可以包括在由数据库界面模块136潜在检索的一个网络响应的所包含的内容。例如，表162和表172可以分别包括公司总量和销售信息。在数据表中的内容可以被一

或多条记录所表示。例如，数据163由记录164和166（而潜在的其他记录由竖向的省略号表示）表示。同样地，数据173可以由记录174和176表示（而潜在的其他记录由竖向的省略号表示）。

数据库界面模块136可以传送命令至数据库141（或其他数据库）从而导致来自数据库141（或其他数据库）的内容被检索。所检索的内容可以用作后来被传送到一发送请求的计算机系统的网络响应的内容。例如，服务器计算机系统111可以从客户端计算机系统101接收网络请求。该网络请求可以被传送到响应结构模块133以处理相应的页面文件从而构建相应的网络响应。响应结构模块133可以检查缓存112以决定该网络响应（或其一部分）是否存储在缓存112之中。当一网络响应存储在缓存112时，网络响应可以直接被返回至客户端计算机系统101。

当网络响应没有存储在缓存112中时，脚本，包括相应的页面文件可以被执行以决定一网络响应。脚本的执行可以包括检查，和如果可用则使用存储在缓存112之中的网络响应。当内容没有存储在缓存112中时，所执行的脚本可以从数据库141中请求内容。因此，对内容的请求被传送到数据库界面模块136中，它可以发布适当的命令以检索变化ID和表ID以及基于变化ID和表ID的被检索的内容。例如，基于变化ID153和表ID152，数据库界面模块136随后可以从表162中检索数据。

也可以这样，版本信息在查询数据之前被检索。例如，数据库界面模块136可以被配置为在查询数据（例如，数据163和/或数据173）之前检索版本信息（例如，一或多个变化ID和一或多个相应的表ID）。在查询数据之前检索版本信息可以明显地减少在数据表中的数据在数据被查询后在版本信息被检索之前被改变这种很少出现的情况的可能性。

响应结构模块133也可以决定所检索的内容是否要被缓存以及所检索的内容是否依赖于—数据表。响应结构模块133可以将要被缓存的所检索的内容，与任何数据表（例如，表ID和相应的变化ID）的依赖性的指示一起，传送到缓存界面模块132。缓存界面模块132可以将所检索的内容插入到缓存112中。因此，后面的该被检索内容的网络请求可以以效率被处理。

所缓存的内容可以存储在表示所缓存内容和相应依赖性的缓存记录数据结构中。缓存记录数据结构的一个域可以是表示在所请求的网络响应中已经（或

将要) 返回客户端计算机的内容的页面内容域。页面内容可以包括网络响应的一部分(例如, 一条或多条来自数据表的记录)或可以包括一完整的网络响应。例如, 响应内容122可以包括来自记录164和166的内容。可替代地, 响应122可以包括包含数据库141中内容的, 来自响应内容134的内容的, 以及来自其他位置(例如, 来自其他数据库或来自其他网络服务)的内容的一完整的网络响应。

缓存记录数据结构的另一个域可以是一个表示识别识别数据库141中数据表的表ID的表ID域。一表ID可以是一个值, 例如, 从一变化通知表中检索到的, 包括数字和/或字母的字符。例如, 当从表162中检索内容时, 表ID152也可以从变化通知表142中被检索。因此, 在缓存112中的一个表ID可以与在变化通知表中的表ID相对应。例如, 当从表172中检索内容时, 表ID128和表ID157都可以识别识别表172。

在一些实施例中, 一个单个的表ID/变化ID元组, 例如, 像这样的, 一个主码记录, 对于每个被监视的数据表来说都存在于缓存中。任何存在于表中的内容都可以依赖于相应的主码记录。因此, 可以这样, 许多个缓存记录依赖于相同的主码记录。当表通过表ID变化来识别识别时, 相应的变化ID可以在主码记录中被更新或者主码将被移除。更新或移除一条主码记录导致任何依赖于缓存的记录(例如, 缓存记录126)被从缓存中移除。使一条单独的主码的从属性无效的多条缓存记录会减少所需查询和版本比较的数量以维持缓存记录。因此, 主码记录可以被用来提高决定何时从属内容无效的效率。

缓存记录数据结构的另一个域可以是表示一由相应表ID识别识别的数据表的变化ID的变化ID域。一个变化ID可以是一个值, 例如, 从变化通知表中被检索到的, 包括数字的字符。例如, 当从表162中检索内容时, 变化ID153可以从变化通知表142中被检索。因此, 在缓存112中的变化ID可以对应于在变化通知表142中的变化ID。例如, 当数据从表172中被检索时, 变化ID129和变化ID158可以相等。因此, 缓存记录数据结构可以是下面的格式:

缓存_记录

{

页面内容

表ID域

变化ID域

}

一条或多条记录数据结构可以被存储在缓存112中。例如，缓存112包括缓存记录121和126。而且，在缓存112中的每列圆点号（也就是，省略号）表示缓存112可以包括的其他缓存记录。在缓存112中的每一条缓存记录对应于包括在前面所构建的网络响应中的部分内容。例如，缓存记录121和126分别包括响应内容122和127，它们可以被包括在传送到客户端计算机系统101的网络响应（或可以是网络响应）中。

有时，服务器计算机系统111可以查询数据库141的版本信息。例如，数据库界面模块136可以被配置成在指定的时间间隔（例如，每五秒钟）查询数据库141。响应于接收的查询，数据库141可以给任何已经从接收的最后一个查询信号后被替换的数据表返回版本信息。一个查询可以是一指定的对数据表的具有区别于包含在该查询中版本信息的版本信息的请求。数据库界面模块136可以接收从数据库141中返回的版本信息。数据库界面模块136将所接收的版本信息传送到缓存界面模块132。缓存界面模块132可以处理所接收的版本信息以识别识别要被无效和从缓存112中移除的缓存记录。可替代地，缓存界面模块132可以处理所接收的版本信息以在缓存112中更新合适的主码记录。

在一替代实施例中，服务器计算机系统111可以查询数据库141中已经变化的版本信息，并且请求该查询不要返回直到一些变化被决定。例如，数据库界面模块136可以被用来给每个被监视的表发布一个阻塞查询。所发布的阻塞查询对于每个受监视的表来说可以包括当前持有的版本信息。在接收所发布的阻塞查询时，数据库141一直等待直到一个或多个被请求的表的版本信息被更新。响应于该更新，数据库141可以返回该变化表的表ID和版本信息。阻塞查询可以被同步执行，从而对数据的请求并不返回直到变化信息可用。另一方面，阻塞查询可以被异步执行，从而请求被立即返回，并且在当变化信息可用时服务器计算机系统111发送一个回复信息。

图2示出了依据本发明原理的一种用于配置一个数据库以提供表格变化通知的方法200一个流程图的例子。可以讨论一下与网络结构100中描述的系统和数据库相关的方法200。方法200包括选择一个数据表的步骤以监视变化的步骤（步骤201）。步骤201可以包括一个对于从要被监视内容变化的数据库中选定的数据表来说是可连接（或包含）的计算机系统。选定的以监视变化的数据表

可以包括数据库界面模块136接收识别要被监视的数据表的用户输入。例如，服务器计算机系统111和/或数据库141的一个管理员可以提交选定用来监视的表选定的162和表172的命令。当一个变化通知表并不总是存在时，管理员也可以输入命令以创建一个变化通知表。例如，服务器计算机系统111和/或数据库141的一个管理员可以提交命令至数据库141以创建变化通知表。

可替代地，计算机系统111可以自动提交选定用来监视数据表的命令，以响应于接收的对于相应包括从所选定的数据表中的内容的网络响应的网络请求。例如，响应于接收一相应包括来自表162和/或表172中内容的网络相应的网络请求，服务器计算机系统111可以自动提交选定用来监视的表162和/或表172选定的命令。当一变化通知表不存在时，服务器计算机系统111可以自动的，例如，基于脚本上的页目录，提交可以导致变化通知表被创建的命令。例如，响应于接收相应包括来自数据库141中数据表的内容的网络响应的网络请求，服务器计算机系统111可以自动提交导致变化通知表142被创建的命令。数据库界面模块136可以传送用户输入和/或自动产生变化通知表创建命令和数据表选择命令至数据库141。

方法200包括插入一条相应于数据表的记录到变化通知表（步骤202）。步骤202可以包括当变化通知表并不总是包括一条相应于数据表的记录时插入一条相应于数据表的记录到变化通知表。例如，与数据库网络互连（或包含）的计算机系统可以插入一条相应于被选定数据表的记录到变化通知表中。一条插入的记录可以包括相应于被选定数据表的版本信息（例如，由版本信息数据结构表示的）。插入到变化通知表的记录可以包括数据库界面模块136接收指示一条将要被插入到变化通知表的记录的用户输入。

例如，服务器计算机系统111和/或数据库141的一个管理员可以提交导致记录151和156将被插入到变化通知表142中的命令。可替代地，计算机系统111可以自动提交导致记录要被插入到相应于接收一相应网络响应的网络请求的变化通知表中的命令，其中该网络响应包括相应选定数据表中的内容。例如，响应于接收对应于包括来自表162和/或表172的内容的网络响应的网络请求，服务器计算机系统111可以自动提交导致记录151和156被插入到变化通知表的命令。数据库界面模块136可以将用户输入或自动生成的记录插入命令传送到数据库141。

方法200包括一个给选定的数据表分配触发器的步骤（步骤203）。步骤203可以包括在当触发器没有分配给该选定的数据表时给选定数据表分配该触发器。例如，与数据库网络互联（或包含）的计算机系统可以分配一个触发器给选定的数据表。一个分配的触发器可以导致当在选定数据表中的内容被替换时选定数据表的版本信息在变化通知表中被更新。例如，当在表162中的内容被替换时（例如，当在数据163中一条记录被插入，移除，或改变时）触发器167会导致变化ID153被更新（例如，增长）。类似地，当在表172中的内容被替换时（例如，当在数据173中一条记录被插入，移除，或改变时）触发器177可以导致变化ID158被更新（例如，增长）。

分配触发器可以包括数据库界面模块136接收指示一个触发器将要被分配到选定的数据表的用户输入。服务器计算机系统111和/或数据库141的一个管理员可以提交导致触发器分配到所选定数据表的命令。例如，管理员可以提交导致将触发器167分配到表162和将触发器177分配到表172的的输入。可替代地，计算机系统111可以自动分配一个响应于接收一相应于包括所选定数据表的内容的网络响应的网络请求。例如，响应于接收一相应于包括来自表162和/或表172的数据的网络响应的网络请求，服务器计算机系统111可以自动提交导致将触发器167分配到表162的和/或导致将触发器177分配到表172的的命令。数据界面模块136可以将用户输入或自动生成的分配命令传送到数据库141。

一个触发器可以包括附属于一个表的部分计算机可执行指令。触发器可以是事件驱动从而当事件发生时该部分计算机可执行指令被执行。例如，触发器167可以被设置为激活当在数据163中的内容被替换时并且可以包括导致变化ID153被递增的部分计算机可执行指令。

方法200包括更新响应于在选定数据表中的被变化的内容的版本信息的步骤（步骤204）。步骤204可以包括与一更新响应于所选定的被改变的数据表的数据库网络互联（或包含）的计算机系统。例如，当数据162中的内容被替换（或者是自动或者是用户输入的结果）时计算机系统可以执行包括导致变化ID153被递增的触发器167的部分计算机可执行指令。

方法200包括返回该已更新的版本信息至请求的服务器计算机系统（步骤205）。步骤205可以包括返回该已更新的版本信息至请求的服务器计算机系统以至于该被更新的版本信息可以被用来决定服务器计算机系统上的缓存记录中

内容的有效性。例如，数据库141可以发送记录151和156（或它们的部分）至服务器计算机系统111以响应于来自服务器计算机系统111的查询。数据库界面模块136可以从数据库141中接收数据记录。数据库界面模块136可以将来自记录151和156的数据（例如，表ID152和157与变化ID153和158）中传送数据至缓存界面模块132以用来识别要被无效的缓存记录。

在一些实施例中，所缓存的内容以对象和结果的形式被用来生成网络响应，例如表格数据，对象，样本，或部分的网络响应。当结果可以被用在超过一个网络响应的的生成中时缓存这些中间结果是合适的，其中每个网络响应包括附加的差异信息（例如，自定义或其他相关信息的结合）。在其他一些实施例中，一个完整的网络响应被缓存。当一个完整的网络响应可以被多个不同网络请求所共享时缓存该完整的网络响应是适当的。当缓存一个完整的网络响应时，该网络响应可能以较低优先级被推迟被缓存在服务器计算机系统中，例如，一个核心模式响应缓存。因此，当该网络响应无效时，它就被从低优先级的缓存中被移除。

图3示出了依据本发明的原理的一个使部分缓存内容无效的方法300的一个流程图的例子。方法300将与在网络结果100中描述的相关计算机和数据库一起讨论。方法300包括查询一相应于一个或多个数据表的版本信息的变化通知表（步骤301）。步骤301可以包括一个服务器计算机系统，它构建包括来自一个或多个数据表中内容的网络响应，在查询变化通知表的相应于一个或多个数据表的版本信息时。例如，数据库界面模块136可以查询变化通知表142的相应于表162和163的版本信息。数据库界面模块136可以查询变化通知表142以作为接收用户输入的结果，指定查询的间隔的出现，响应于一个检索所缓存内容的请求，或一个优先于来自依赖表中请求的数据初始化的阻塞查询。

一个查询可以被配置为请求所有被监视的在变化通知表中显示的数据表的版本信息，请求用于具有突出依赖性的被监视的数据表的版本信息，或者请求用于已经改变（例如，通过包括在查询中的当前版本信息）被监视的数据表的版本信息。响应于一个查询，更新的版本信息从数据库返回到服务器计算机系统。基于查询的配置，数据库可以返回，例如，所有被监视数据表的版本信息，具有突出缓存依赖性的被监视的数据表的版本信息，或者已经改变的被监视的数据表的版本信息。

可替代地，服务器计算机系统可以发布一个查询至变化通知表并请求被发布的查询阻塞在服务器计算机系统上直到所请求表的版本信息被更新。例如，服务器计算机系统可以发布查询，该查询被阻塞在服务器计算机系统上直到当前的版本信息大于一指定的阈值。所发布的阻塞的查询可以被同步执行，从而使服务器计算机系统上的一个相应的线程被阻塞直到结果是可用的。另一方面，所发布的阻塞的查询可以被异步执行，从而使相应的在服务器计算机系统上的一个线程（例如，在一独立的线程）不被阻塞并且在服务器计算机系统上的结果可用时调用一个回复信息。

方法300包括一个接收至少一个来自变化通知表被最后查询时就已经改变的数据表的列表的步骤302。步骤302可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的网络响应，接收来自至少一个自从变化通知表被最后查询时就已经改变的数据表的列表的步骤。至少一个数据表的列表可以包括该至少一个数据表的当前版本信息。例如，数据库141可以返回包括数据表162的版本信息（例如，表ID152和变化ID153）以及表172的版本说明的数据表（例如，表ID157和变化ID158）的版本信息到服务器计算机系统111。数据库界面模块136可以接收从数据库141返回的数据表的一个列表。

方法300包括一个决定所缓存的内容依赖于至少一个数据表的步骤（步骤303）。步骤303可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容的网络响应，决定所缓存的内容依赖于至少一个数据表。例如，服务器计算机系统111可以决定依赖于表162的响应内容122和依赖于表172的响应内容127。缓存界面模块132可以将所接收的表ID（表ID152和157）与相应的缓存的表ID（表ID123和128）相比较。当表ID的比较如此说明（例如，当所接收的表ID和所缓存的表ID识别属于相同的数据表时）时，缓存界面模块132决定依赖于一个数据表所缓存的内容。例如，当表ID123的值与表ID152的值匹配时，缓存界面模块132可以决定依赖于表162响应内容。类似地，当表ID128的值与表ID157的值匹配时，缓存界面模块132可以决定依赖于表172响应内容127方法300包括当前版本信息与所缓存的版本信息的比较步骤（步骤304）。步骤304可以包括一个服务器计算机系统，它包括构建包括来自至少一个数据表的内容的网络响应，比较当前的版本信息和所缓存的版本信息。例如，服务器计算机系统111可以将从返回的列表中的版本信息与存储在缓存112中的版本信息相比较。

缓存界面模块132可以将所接收的变化ID（变化ID153和158）与相应的所缓存的变化ID（变化ID124和129）相比较。

方法包括移除基于比较结果的部分所缓存内容的步骤（步骤305）。步骤305可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容的网络响应，移除基于比较结果的所缓存的部分内容。例如，服务器计算机系统111可以从缓存112中移除所缓存的基于当前版本信息与所缓存的版本信息的比较。当变化ID的比较这样指示（例如，当一个所接收的变化ID的值比所缓存的变化ID的值大时），缓存界面模块132移除缓存的内容。例如，当变化ID153比变化ID124大时，缓存界面模块132可以从缓存112中移除缓存记录121。类似地，当变化ID158比变化ID129大时，缓存界面模块132可以从缓存112中移除缓存记录126。当变化ID相等时所缓存内容不会从缓存112中被移除。

因此，通过使用数据库缓存依赖性，适当的使所缓存内容无效的可能性会增加。当数据表没有改变时，从缓存中预先完全移除无效的依赖内容的机会会增加。因此，来自缓存的无效依赖内容可以被用来更为有效的构建网络响应。另一方面，当数据表已经被改变，从缓存中移除相应无效依赖内容的机会会增加。因此，无效依赖内容潜在地阻止被包含在网络响应中。

图4示出了依据本发明原理的一种用于访问请求客户端计算机系统显示内容的方法400的一个流程图的例子。方法400将与在网络结构100中描述的相关的计算机系统和数据库一起讨论。方法400包括从一客户端计算机系统接收网络请求的步骤（步骤401）。步骤401可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容的网络响应，从一客户端接收网络请求。例如，服务器计算机系统111可以从客户端计算机系统101（例如，在浏览器102接收的作为用户输入的结果发起的）接收网络请求。一个网络请求可以推导，例如，当浏览器102访问一个相应于网络响应的通用资源定位器（“URL”）时。

一个网络请求可以请求包括被包含在一数据表中的所请求的内容的网络响应的递送。例如，一个来自客户端计算机系统101的网络响应可以是一个包括来自表162和/或表172的网络响应的请求。表162和表172通常可以访问其他的可以替换在数据表162和172（包括将要被包括在所请求的网络响应中的内容）中被包含的内容的应用（例如，工作流程或应用总量）。因此，可以这样，包括在数据163和数据173中的内容通过在网络请求之间的其它应用被替换。

响应于接收一网络请求，服务器计算机系统111可以决定一个完整的相应网络响应被存储在缓存112中（或在一低优先级缓存中）。当该完整的相应网络响应被缓存时，服务器计算机系统111可以从缓存112中返回该完整的被缓存的网络响应而不需进一步请求任何作为网络请求的结果的处理。因此，当一完整的网络响应被缓存时，与一网络响应的完成相关的潜在性可以明显的减少。

另一方面，响应于接收一网络请求，服务器计算机系统111可以决定至少一部分的相应网络响应当前不被缓存。方法400包括一个决定要被包括的相应网络响应的内容的至少一部分当前不被缓存（步骤402）。步骤402可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容的网络响应，决定包括在网络响应中的至少一部分内容当前没有被缓存到服务器计算机系统。例如，服务器计算机系统111可以决定包括在将被递送到客户端计算机系统101的网络响应中的至少一部分内容当前没有被缓存在计算机系统111。

一个在服务器计算机系统111接收的网络请求可以被传送到响应结构模块133。因此，响应结构模块133可以检索一个适当的页面文件（例如，一个ASP文件）和处理该页面文件以构建一相应的网络响应。包括在该页面文件中的脚本可以请求存储在服务器计算机系统（例如，在表162和/或163中）外部位置的内容。当一个外部内容的请求被处理时，响应结构模块133可以发送适当的指令至缓存界面模块132以致使缓存界面模块132尝试查找在缓存112中的外部内容的地点。当外部内容被存储在缓存112中时，缓存界面模块132可以响应说明外部内容存储在缓存112中的响应结构模块。另一方面，当外部内容没有存储在缓存112中时，缓存界面模块132可以响应说明外部内容没有存储在缓存112中的响应结构模块。

当决定了包括在没有被缓存的一个网络响应的至少一部分内容时，服务器计算机系统111可以进一步决定该至少一部分内容是否依赖于数据库141中的任何数据表。当该至少一部分内容依赖于数据库141中的一个数据表时，服务器计算机系统111可以进一步决定数据表的一条相应主码记录是否存储在缓存112中。当相应的主码记录不是存储在缓存112中时，服务器计算机系统111可以检索数据表（例如，来自变化通知表142）的适当的版本信息并将一个包括适当版本信息相应的主码记录存储在缓存112中。

方法400包括一个访问来自数据表请求的内容的步骤（步骤403）。步骤403

可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容的网络响应，来自访问数据表中所请求的内容。例如，服务器计算机系统111可以从数据库141的一个表中访问所请求的内容。作为不存储在数据库141中的所请求的内容的指示的响应，响应结构模块133可以发送适当的指令（例如，通过发布一个查询）至数据库界面模块136以致使数据库界面模块136尝试找到数据库141中所请求内容的位置。数据库141可以寻找所请求内容（例如，在记录164，166，174，和/或176中）的位置并返回所请求的内容至数据库界面模块136。数据库界面模块136然后可以传送所请求的内容至响应结构模块133。

方法400包括构建相应网络响应的步骤，该网络响应包括所请求的内容（步骤404）。步骤404可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容的网络响应，并构建该网络响应。例如，服务器计算机系统111可以包括在一要被返回到客户端计算机系统101的网络响应中的来自数据库141的内容。响应结构模块133可以包括从数据库界面模块136接收的所请求的内容（例如，来自记录164，166，174，和/或176）以满足一网络请求。基于所请求的内容（与潜在的包括在一适当页面文件中的在其他HTML和XML命令以及脚本一样），响应结构模块133可以生成一个适当的网络响应。

方法400包括用于存储所请求内容以至于所请求的内容可以被有效的访问以满足后续的网络请求的面向结果的功能步骤（步骤407）。步骤407可以包括任何用于存储所请求的内容的相应步骤从而使所请求的内容被有效的访问以满足后续的网络请求。然而，在图4的方法的例子中，步骤407包括一给所构建的网络响应的至少一部分分配数据库缓存依赖性的相应步骤（步骤405）。步骤405可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容的网络响应，给所构建的网络响应的至少一部分分配数据库缓存依赖性。

步骤407也包括一个缓存至少一部分所构建的网络响应的相应步骤（步骤406）。步骤406可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容，缓存至少一部分所构建的网络响应，服务器计算机系统可以在服务器计算机系统的缓存位置缓存至少一部分所构建的网络响应。例如，缓存界面模块132可以导致由响应结构模块133构建的网络响应的至少一部分优先于所构建的网络响应被递送到客户端计算机系统而被存储在缓存112中。

当缓存一个完整的网络响应时，该网络响应可以被推迟被缓存在一服务器

计算机系统的低优先级存储器上，例如，在一核心模式响应缓存中。例如，一个完整的网络响应可以被发送到一由因特网信息服务器（“IIS”）模块维持的缓存位置。因此，当该网络响应无效时，就被从低优先级缓存中被移除。

数据库缓存依赖性可以被分配并且所缓存内容的至少一部分可以作为在网络相应构建过程中所执行命令的结果。在一些实施例中，数据库缓存依赖性被分配给一个网络响应中的部分内容。下面的第一个例子的指令可以被包括在一页面文件中以导致在网络响应的部分内容被分配一数据库缓存依赖性：

```
1. <HTML>
2.     <Script>
3.         SqlCacheDependency c;
4.         c=new SqlCacheDependency(pubs,products);
5.         ProductCollection p=getproducts();
6.         Cache.Insert("Pinfo",p,c);
7.     </Script>
8. </HTML>
```

这个例子的第3行的指令创建一个数据类型为SqlCacheDependency的变量“c”。SqlCacheDependency可以是一个已经被预先建立的用于表示数据库缓存依赖性的数据类型。这个例子的第4行的指令分配该变量c以表示包含在“pubs”数据库中的表“products”的依赖性。这可以表示，例如，包含在数据库141上的表162的依赖性。这个例子的第5行的指令创建一个数据类型为“ProductCollection”的变量“p”。ProductCollection可以是一个预先被建立的用于表示从产品数据库中接收的产品列表的数据类型。函数“getproducts()”在第5行被执行以返回一个存储在变量p里的产品列表（例如，从表162和/或163中）。

这个例子的第6行的指令创建一条缓存记录（例如，在缓存112中）。缓存记录用主码“Pinfo”创建，随后它可以被用做缓存记录的参考。该缓存记录被创建并存储在由变量p表示的产品列表中（例如，作为响应内容122）并且缓存记录依赖于在pubs数据库中的用变量c（例如，作为表ID123）表示的产品表。当大量的网络响应都包括相同部分的内容时第一个例子中的指令可以是特别有利的。

下面例子中的指令可以被包括在一个页面文件中以检索所缓存的产品列表：

ProductCollection p= Cache (“Pinfo”);

当检测到在pubs数据库中的产品表被替换时，通过主码Pinfo所参考的缓存记录是无效的。在其他一些实施例中，一个数据库缓存依赖性被分配给一完整的网络响应。下面的第二个例子的指令可以被包含在一页面文件中导致一完整的网络响应被分配给一数据库缓存依赖性：

1. <%Output Cache SqlTableDepend= “pubs:products”%>
2. <HTML>
3. <Script>
4. ...
5. </Script>
6. </HTML>

第二个例子的第1行指令是导致该从网页文件产生的依赖于包含在“pubs”数据库中的“products”表的完整网络响应的指示。一条缓存记录可以被创建以缓存该从网页文件产生的完整的网络响应（例如，作为响应内容127）。在第4行的省略号表示其他可以执行的脚本指令，例如，像这样的，当页面文件被处理时，访问在pubs数据库里的产品表的。当大量客户端请求相同的网络响应时第二个例子的指令可以是很有利的。

数据库缓存依赖性被结合一个或多个其他缓存依赖性使用也是可以的，例如，像这样的，一时间，文件，主码，和/或网络服务缓存依赖性或者甚至其他数据库缓存依赖性。例如，下面的例子的指示可以被包含在一页面文件中导致一完整的网络响应被分配到不光数据库缓存依赖性还有时间依赖性：

<%Output Cache Duration=600s SqlTableDepend= “pubs:customers” %>

该例子的指示会导致从相应网络响应被创建或直到该在“pubs”数据库中的“用户”表被替换时在缓存中要维持600秒的相应网络响应。

在一些实施例中，一或多条缓存记录是关联于（或者，与之有联系的）数据库依赖缓存记录的（因此产生一个主码依赖性）主码依赖缓存记录。因此，使数据库依赖缓存记录无效会导致每一个主码依赖缓存记录也无效。因为主码依赖缓存记录是依赖于在缓存中（例如，在系统存储器中）的其他数据的，使一主码依赖缓存记录无效是比使一数据库依赖缓存记录（它可以要求一数据库查询，网络通信，数据串行化，等等）无效更有效。因此，为了增加使缓存记

录的无效的效率，，一个具有依赖内容的第一缓存记录可以被配置为依赖于—第二缓存记录（通过—主码依赖性）。当在相应数据表中的变化被检测时，该第二缓存记录使任何依赖缓存记录无效。响应于—检测到的变化，该第二缓存记录也可以从存储或者适当的用于可以被更新的相应的数据表的版本信息中被移除。

例如，缓存记录121和缓存记录126都依赖于表162（例如，表ID123和表128都对应于表ID152）也是可以的。为了优化决定何时响应内容122和响应内容127被无效的效率，一条依赖于表162的主码记录可以被插入到缓存112中（也就是，插入到系统存储器中）。缓存记录121和126可以依赖于所插入的主码记录。当表ID162被替换时，数据库141的一个后续查询可以识别变化ID153被更新并使所插入的主码记录无效。使所插入的主码记录无效会自动使缓存记录121和126无效而不是要求变化ID124和变化ID129直接与变化ID153进行比较。因此，数据库查询的次数和版本信息可以被减少并且网络，数据库，和处理资源得以保留。

方法400包括一个将所构建的网络响应传送至响应于网络请求的客户端计算机系统的步骤（步骤408）。步骤408可以包括一个服务器计算机系统，它构建包括来自至少一个数据表的内容的网络响应，将所构建的网络响应传送至响应于网络请求的客户端计算机系统。例如，响应于该来自客户端计算机系统101的网络请求服务器计算机系统111可以递送—网络响应至客户端计算机系统101。所递送的网络响应可以是一个由响应结构模块133构建的网络响应。

图5示出了一种用于本发明原理的适合的操作环境。图5以及下面的论述会提供实施本发明涉及的适合的计算环境一个简洁，通用的描述。虽然没有被要求，该环境仍然以用计算机可执行指令通用格式被描述，例如程序模块可以被计算机系统执行。一般来说，程序模块包括例程序，程序，对象，组件，数据结构，以及其类似，其执行特殊任务或执行特别的抽象数据类型。计算机可执行指令，相关的数据结构，和程序模块表示用于执行这里公开的方法中的步骤的程序代码方式的例子。

参照图5，一个用于实行本发明的示范性系统可以包括—在计算机系统520形式的通用计算机设备，包括—处理单元521，—系统存储器522，和—系统总线523用于将包括系统存储器522的多种系统组件相连至处理单元521。处理单元521可以执行设计用来实现计算机系统520特点的计算机可执行指令，包括本发

明中的特点。系统总线523可以是多种总线结构类型中的任何一种，包括一存储器总线或存储控制器，一外设总线，和一使用各种总线结构中任一种的本地总线。系统存储器包括只读存储器（“ROM”）524和随机存储器（“RAM”）525。一基本输入/输出系统（“BIOS”）526，它包括基本的例程式有助于在计算机520中的组件之间传送信息的事务，例如在启动期间，可以存储在ROM524中。

计算机系统520也可以包括一个用来从硬磁盘539读出和写入的硬磁盘驱动器527，一用来从可移动磁盘529读出和写入的磁盘驱动器528，以及一用来从一可移动光盘531中读出或写入的光学磁盘驱动器630，例如，或者是，一CD-ROM或其他光学媒体。硬磁盘驱动器527，磁盘驱动器528，以及光学磁盘驱动器530都通过硬磁盘驱动器接口552、磁盘驱动器接口533和光学磁盘驱动器接口534与系统总线523分别相连。这些驱动器和它们相关的计算机可读媒体提供计算机可执行指令、数据结构、程序模块、以及其它用于计算机系统520的其他数据的非易失性的存储。虽然这里描述的本发明的实例环境使用了一个硬磁盘539，一可移动磁盘529和一可移动光盘531，也可以使用用来存储数据的其他类型的计算机可读媒体，包括磁带，闪存卡，数字通用磁盘，Bernoulli磁盘，RAM，ROM，及其类似。

程序代码方法包括一个或多个可以存储在硬盘539、磁盘529、光盘531、ROM524或RAM525中的程序模块，包括一个操作系统535，一个或多个应用程序536，其他程序模块537，和程序数据538。一个用户可以通过键盘540，定位设备542，或其他输入设备（未示出）给计算机系统520输入命令或信息，例如，像这样的，一个话筒，游戏手柄，游戏控制台，扫描仪，或其类似。这些和其他输入设备可以通过与系统总线523耦合的串行接口546被连接到处理单元521。可替代地，输入设备可以通过其他接口被连接，例如，像这样的，一并行端口，一游戏端口，一通用串行总线（“USB”）端口，或一带电线的端口。一监视器547或其他显示设备也通过视频适配器548与系统总线523相连。计算机系统520也可以连接到其他外部输出设备（未示出），例如，像这样的，扬声器和打印机。

计算机系统520和网络是可互连的，例如，像这样的，一个办公室范围或企业范围的计算机网络，一内部网，和/或互联网。计算机系统520可以与外部资源

交换数据，例如，像这样的，通过这种网络连接的远程计算机系统，远程应用程序，和/或远程数据库。

计算机系统520包括网络接口553，通过它计算机系统520从外部资源接收数据和/或传送数据至外部资源。如图5所述，网络接口553使通过链接551与远程计算机系统583交换数据更为容易。链接551表示网络的一部分。而远程计算机系统583表示网络中的一个节点。例如，远程计算机系统583可以维持一给为在计算机系统520所构建的网络响应提供内容的数据库。另一方面，计算机系统583可以是一个从一计算机系统520所维持的数据库中请求内容的服务器计算机。

同样地，计算机系统520包括串行接口546，通过它计算机系统520从外部资源接收数据和/或传送数据至外部数据。串行接口546耦合至调制解调器554，通过它计算机系统520从外部资源接收数据和/或传送数据。可替代地，调制解调器554可以是一个通过适当接口与计算机系统520相连的符合电缆服务接口规格的数据（“DOCSIS”）调制解调器或数字用户线（“DSL”）调制解调器。然而，如图5所描述的，串行接口54和调制解调器554使通过链接552与远程计算机系统593的数据交换更为容易。链接552表示网络的一部分，而远程计算机系统表示网络中的一个节点。例如，远程计算机系统593可以是一个从计算机系统520请求网络响应的客户端计算机系统。另一方面，计算机系统593可以是一个给计算机系统520提供网络响应的服务器计算机系统。

而图5表示一个用于本发明的适合的操作环境，本发明的原理在该系统中可以得到体现，如果必要可进行适当的修改，以实现本发明的原理。图5所示的环境仅仅是示范性的决不是表示在本发明原理可以实现的环境中的很大范围中的一小部分。

本发明的模块，除了相关的数据外，可以从任何与计算机系统520相关的计算机可读媒体被存储并访问。例如，部分的这些模块和部分相关的程序数据可以被包括在操作系统535，应用程序536，程序模块537和/或程序数据538中，用于系统存储器522的存储。当一大容量存储设备，例如，像这样的，硬磁盘539，与计算机系统520耦合时，这些模块和相关程序数据也可以被存储在该大容量存储设备中。在一个网络环境中，被描述为与计算机系统520相关的程序模块和相关数据，或者它们的部分，可以被存储在远程存储设备中，例如，像这样的，与远程计算机系统583和/或远程计算机系统593相关的系统存储器和/或大容量

存储设备。这些模块的执行可以在一如前所述的分布式环境中进行。

本发明可以在不脱离其精神和本质特点的情况下用其它的指定方式实现。所描述的实施例是在所有方面仅仅用于示例性的而不是限制性的。因此，本发明的范围是，由所附的权利要求书来说明而不是前面的说明书。所有的在权利要求范围内的等效的改变都是包含在其范围内的。

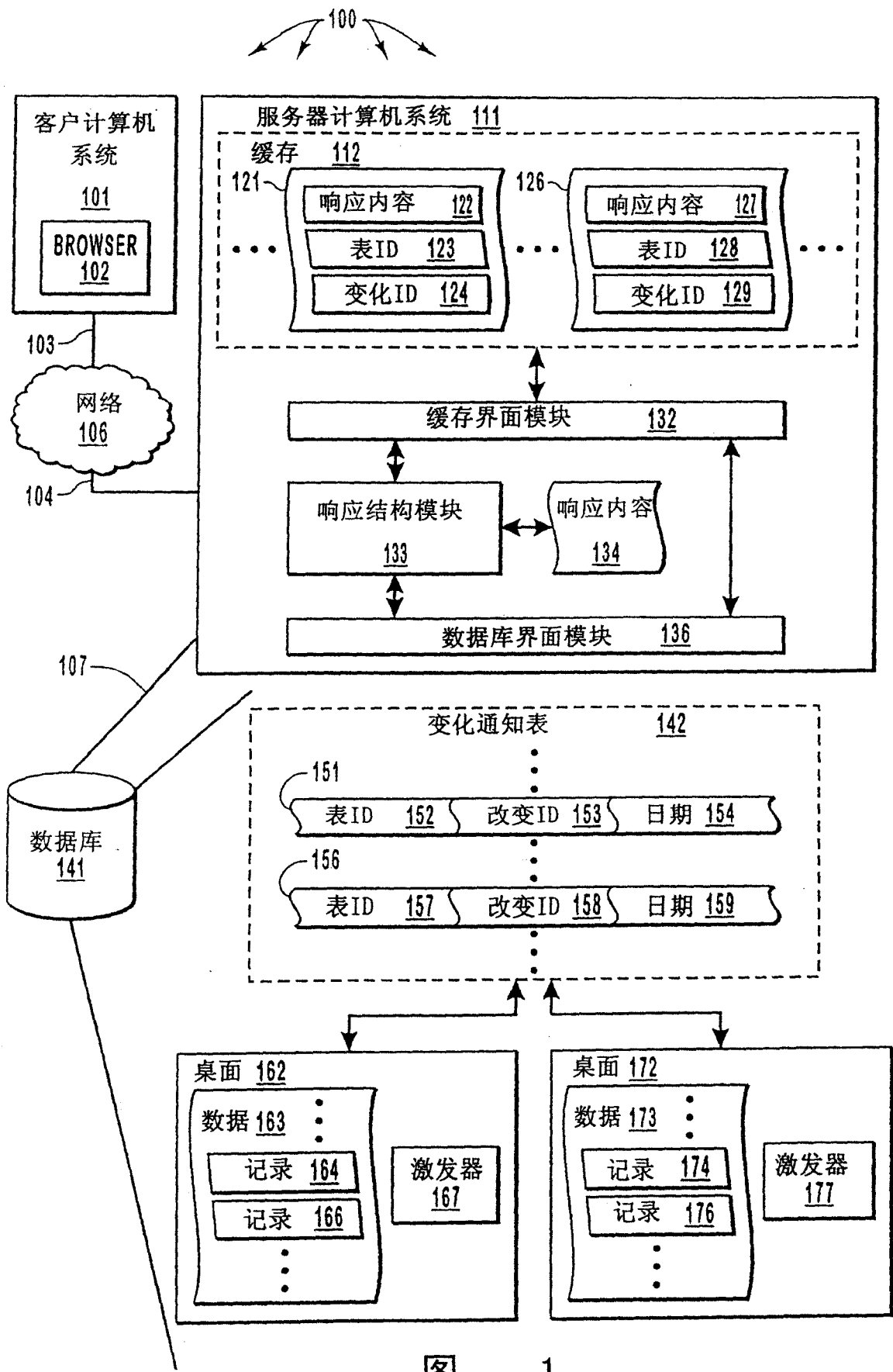


图 1

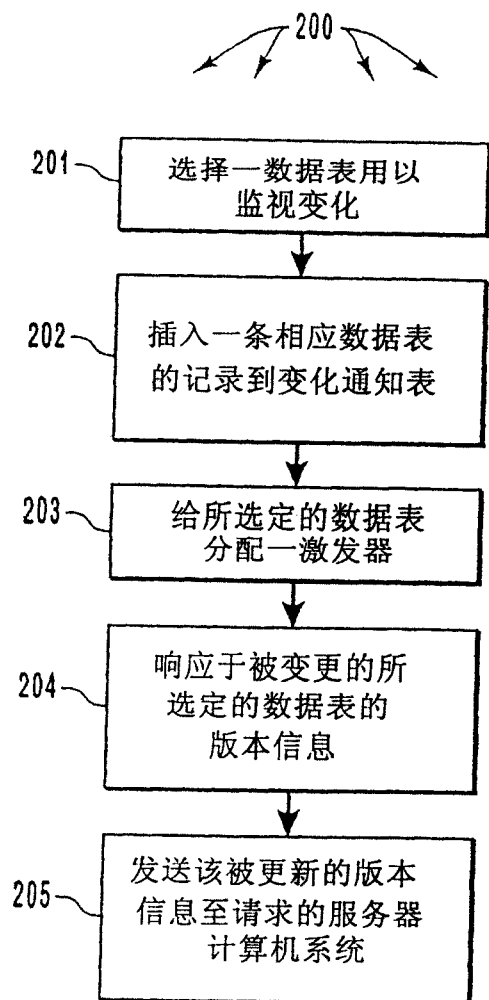


图 2

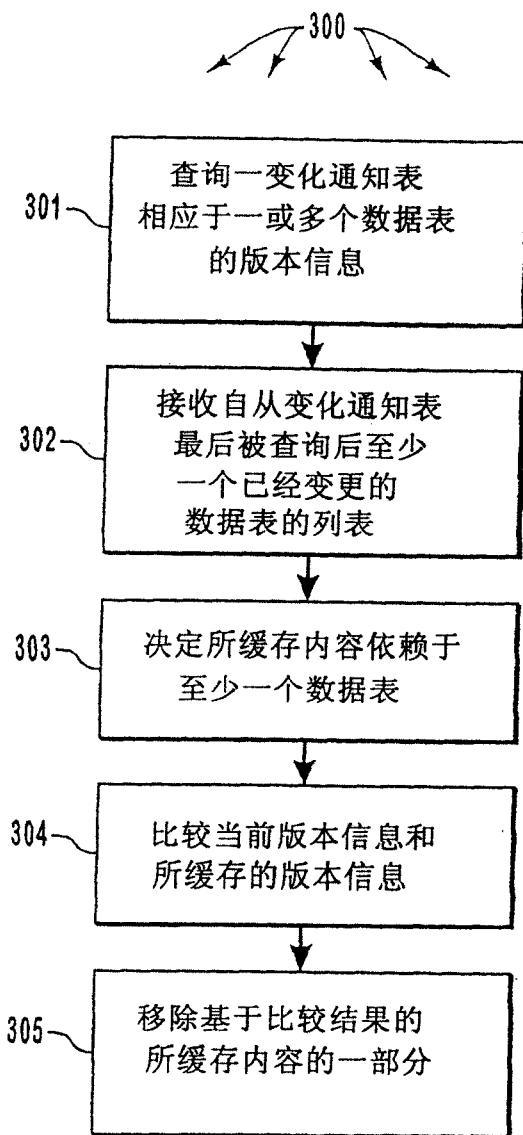


图 3

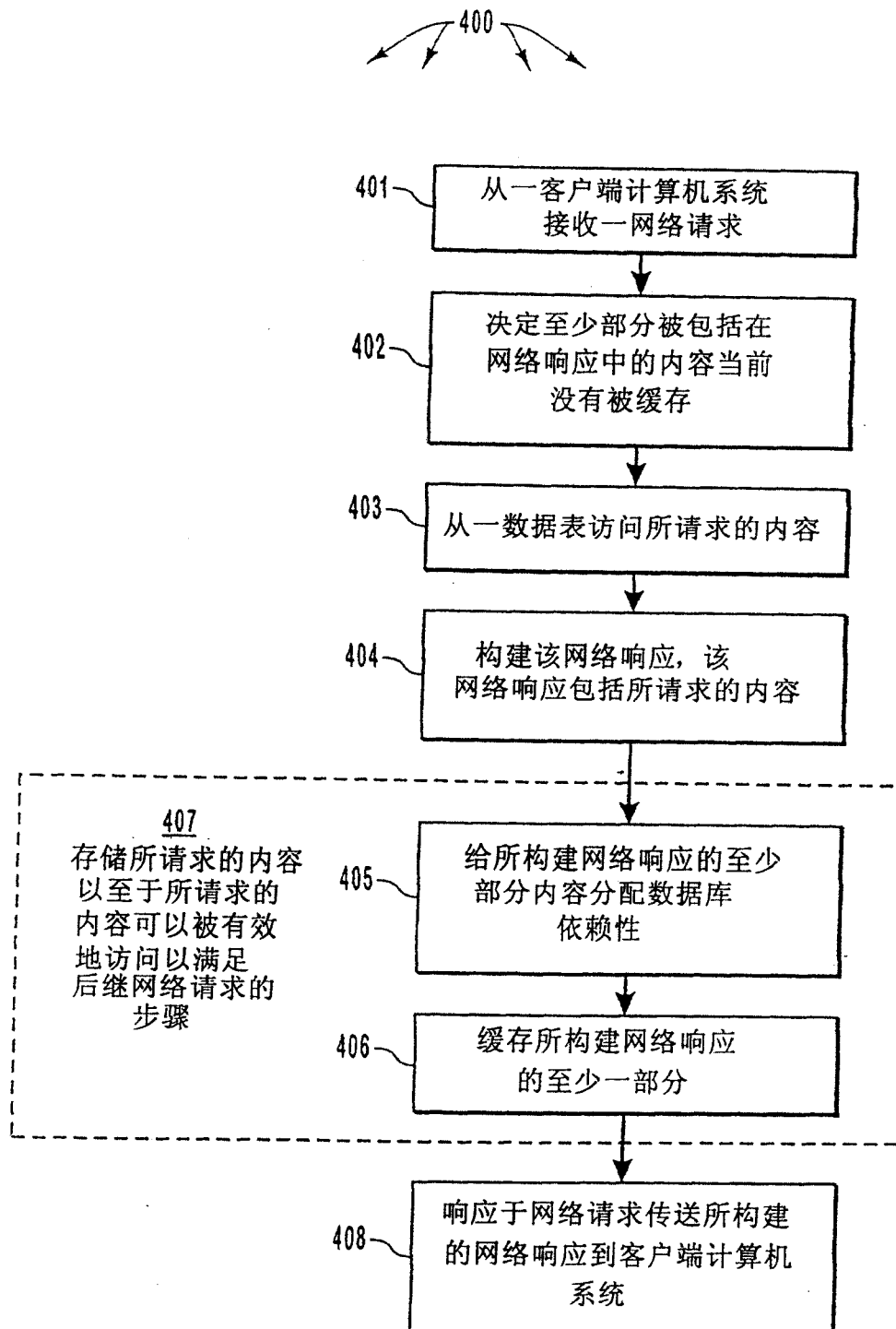


图 4

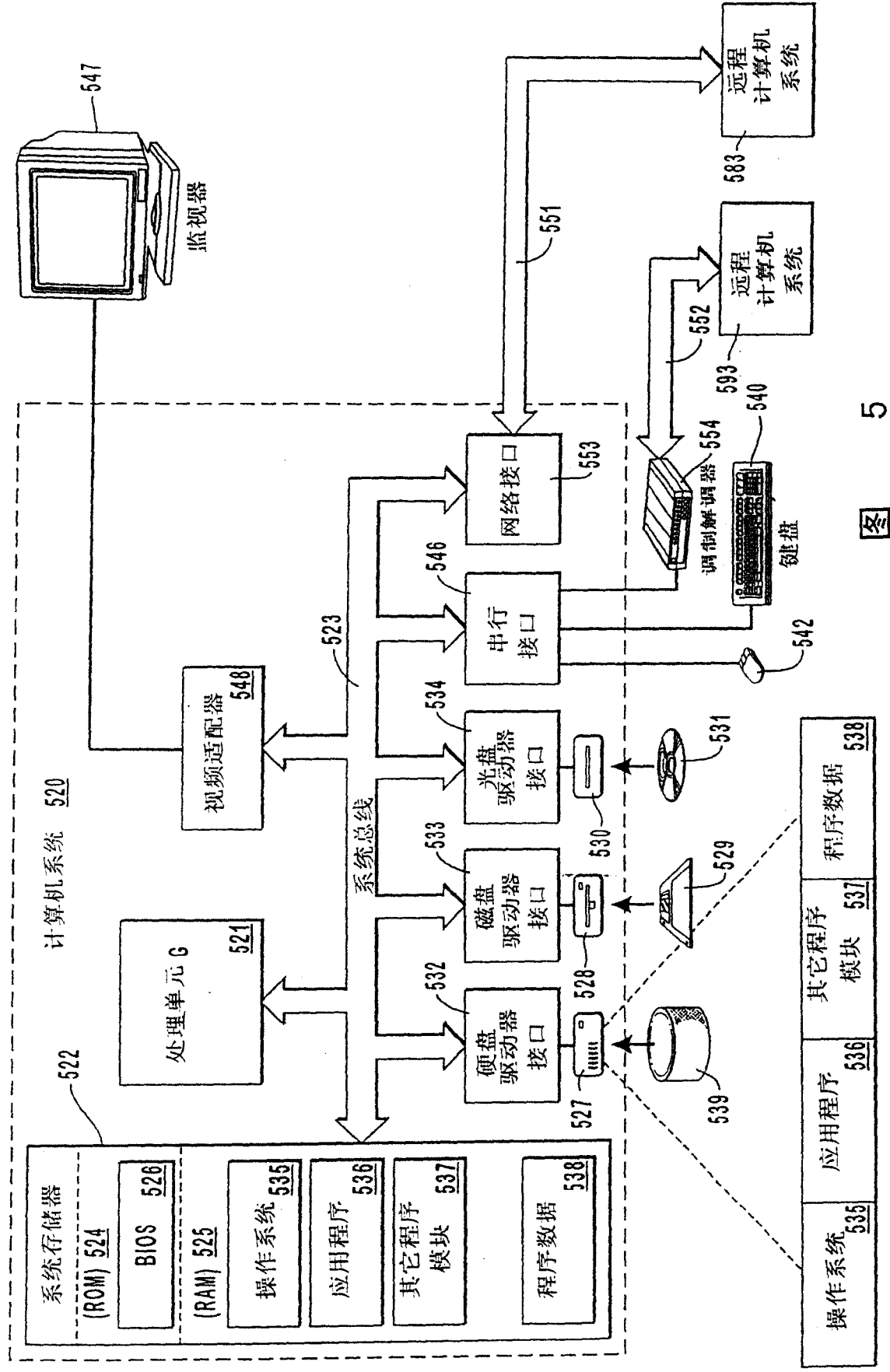


图 5