



(12) 发明专利

(10) 授权公告号 CN 101390094 B

(45) 授权公告日 2011.04.13

(21) 申请号 200780006163.9

G06F 15/16(2006.01)

(22) 申请日 2007.01.26

(56) 对比文件

(30) 优先权数据

US 2005/0177617 A1,2005.08.11,
 CN 1564989 A,2005.01.12,
 CN 1520562 A,2004.08.11,
 US 6983293 B2,2006.01.03,
 CN 1479223 A,2004.03.03,

(85) PCT申请进入国家阶段日

2008.08.21

审查员 杨盈霄

(86) PCT申请的申请数据

PCT/US2007/002186 2007.01.26

(87) PCT申请的公布数据

WO2007/106239 EN 2007.09.20

(73) 专利权人 微软公司

地址 美国华盛顿州

(72) 发明人 S · 托比斯

(74) 专利代理机构 上海专利商标事务所有限公

司 31100

代理人 陈斌

(51) Int. Cl.

G06F 17/30(2006.01)

G06F 17/40(2006.01)

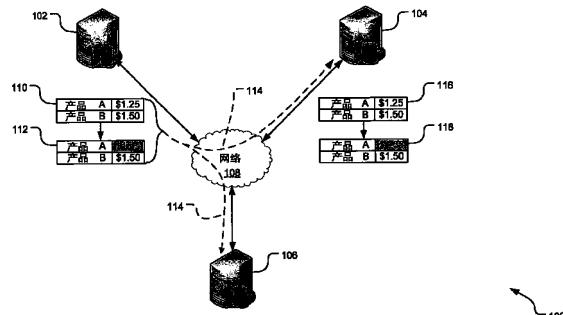
权利要求书 2 页 说明书 7 页 附图 4 页

(54) 发明名称

针对复制数据库的分布式冲突解决

(57) 摘要

响应于对一个或多个同步冲突的检测而和每个复制品的同步过程并发运行一个或多个解决过程。每个解决过程用于解决有关数据对象的冲突而同步过程则继续处理其他同步更新。在冲突解决已被确定之后，针对本地复制品的当前条件测试冲突解决。如果冲突解决不陈旧，则将其应用于本地复制品并传递回同步过程以供传播至远程复制品。否则，如果冲突解决陈旧，则被丢弃。



1. 一种在分布式数据库的本地复制品同步中解决冲突的方法，所述方法包括：

执行从所述分布式数据库的一个或多个远程复制品接收同步更新的同步过程，如果在所述本地复制品的本地数据对象和同步更新之间检测到冲突，则所述同步过程初始与所述本地数据对象相关联的解决过程；以及

响应于由所述同步过程初始的所述解决过程，和所述同步过程并发执行所述解决过程，所述解决过程确定冲突解决而所述同步过程则处理其他同步更新，其中如果在所述本地数据对象和所述同步更新之间检测到冲突，所述同步过程还记录所述冲突的快照并将所述快照传递给所述解决过程，并且其中所述解决过程还通过将所述快照与所述冲突解决已被确定之后存在的条件相比较来确定所述冲突解决是否陈旧。

2. 如权利要求1所述的方法，其特征在于，所述解决过程还用所述冲突解决更新所述本地数据对象，而所述同步过程则处理其他同步更新。

3. 如权利要求1所述的方法，其特征在于，所述解决过程还确定所述冲突解决是否陈旧，并且如果所述冲突解决不陈旧则用所述冲突解决更新所述本地数据对象。

4. 如权利要求1所述的方法，其特征在于，所述快照在所述冲突解决之前被捕捉并且包括本地时间戳、所述本地数据对象的状态、以及从所述同步更新接收的相应远程数据对象的状态。

5. 如权利要求1所述的方法，其特征在于，所述解决过程还记录与所述本地数据对象相关联的同步参数并且如果来自所述快照的所述本地时间戳早于所述同步参数，则确定所述冲突解决是陈旧的。

6. 如权利要求1所述的方法，其特征在于，所述解决过程还将所述冲突解决传递给所述同步过程用以传播至一个或多个远程复制品。

7. 如权利要求1所述的方法，其特征在于，如果没有在所述本地数据对象和同步更新之间检测到冲突，则所述同步过程将所述同步更新应用于所述本地数据对象。

8. 如权利要求1所述的方法，其特征在于，所述解决过程还在用所述冲突解决更新所述本地数据对象之前锁定所述本地数据对象。

9. 如权利要求1所述的方法，其特征在于，所述解决过程还在测试所述冲突解决是否陈旧之前锁定所述本地数据对象。

10. 如权利要求1所述的方法，其特征在于，如果所述冲突解决陈旧，则所述解决过程还删除所述冲突解决。

11. 一种在分布式数据库的本地复制品同步中解决冲突的系统，所述系统包括：

用于执行从所述分布式数据库的一个或多个远程复制品接收同步更新的同步过程的装置，如果在所述本地复制品的本地数据对象和同步更新之间检测到冲突，则所述同步过程初始与所述本地数据对象相关联的解决过程；以及

用于，响应于由所述同步过程初始的所述解决过程，和所述同步过程并发执行所述解决过程的装置，所述解决过程确定冲突解决而所述同步过程则处理其他同步更新，其中如果在所述本地数据对象和所述同步更新之间检测到冲突，所述同步过程还记录所述冲突的快照并将所述快照传递给所述解决过程，并且其中所述解决过程还通过将所述快照与所述冲突解决已被确定之后存在的条件相比较来确定所述冲突解决是否陈旧。

12. 一种在分布式数据库的本地复制品同步中解决冲突的方法，所述方法包括：

执行从所述分布式数据库的一个或多个远程复制品接收同步更新的同步过程，如果在所述本地复制品的本地数据对象和同步更新之间检测到冲突，则所述同步过程初始与所述本地数据对象相关联的解决过程；以及

响应于由所述同步过程初始的所述解决过程，和所述同步过程并发地执行所述解决过程，所述解决过程确定冲突解决并且如果所述本地数据对象自从所述解决过程初始起尚未被所述同步过程更新，则将所述冲突解决应用于所述本地数据对象。

13. 如权利要求 12 所述的方法，其特征在于，所述解决过程还用所述冲突解决更新所述本地数据对象，而所述同步过程则处理其他同步更新。

14. 如权利要求 12 所述的方法，其特征在于，所述解决过程还确定所述冲突解决是否陈旧，并且如果所述冲突解决不陈旧则用所述冲突解决更新所述本地数据对象。

15. 一种在分布式数据库的本地复制品同步中解决冲突的系统，所述系统包括：

用于执行从所述分布式数据库的一个或多个远程复制品接收同步更新的同步过程的装置，如果在所述本地复制品的本地数据对象和同步更新之间检测到冲突，则所述同步过程初始与所述本地数据对象相关联的解决过程；以及

用于，响应于由所述同步过程初始的所述解决过程，和所述同步过程并发地执行所述解决过程的装置，所述解决过程确定冲突解决并且如果所述本地数据对象自从所述解决过程初始起尚未被所述同步过程更新，则将所述冲突解决应用于所述本地数据对象。

16. 一种在分布式数据库的本地复制品同步中解决冲突的方法，所述方法包括：

执行从所述分布式数据库的一个或多个远程复制品接收同步更新的同步过程，如果在所述本地复制品的本地数据对象和同步更新之间检测到冲突，则所述同步过程在分开的线程中产生解决过程；以及

和所述同步过程同时执行所述分开线程中的所述解决过程，所述解决过程确定冲突解决同时所述同步过程处理其他同步更新。

17. 如权利要求 16 所述的方法，其特征在于，所述解决过程还用所述冲突解决更新所述本地数据对象，而所述同步过程则处理其他同步更新。

18. 如权利要求 16 所述的方法，其特征在于，所述解决过程还确定所述冲突解决是否陈旧，并且如果所述冲突解决不陈旧则用所述冲突解决更新所述本地数据对象。

19. 一种在分布式数据库的本地复制品同步中解决冲突的系统，所述系统包括：

用于执行从所述分布式数据库的一个或多个远程复制品接收同步更新的同步过程的装置，如果在所述本地复制品的本地数据对象和同步更新之间检测到冲突，则所述同步过程在分开的线程中产生解决过程；以及

用于和所述同步过程同时执行所述分开线程中的所述解决过程的装置，所述解决过程确定冲突解决同时所述同步过程处理其他同步更新。

针对复制数据库的分布式冲突解决

[0001] 背景

[0002] 一般而言，数据库是以系统性的方式记录在存储介质中以供计算系统访问的经组织的数据对象（例如，数据表、数据记录、文件等）集合。每个数据对象可以包括通常被组织为一组键标数据元素或值以便检索和排序的一个或多个数据记录。数据库一般由模式结构化地描述，而模式指定数据库中数据对象的类型和关系。诸如数据库管理系统 (DBMS) 之类的程序能够查询数据库以访问指定的数据对象。

[0003] 复制允许将一个存储介质的单个数据库拷贝到第二存储介质，从而有效创建原始数据库的独立实例。第二存储介质上的数据库副本被称为“复制数据库”。原始数据库和复制品（统称为“复制品”）共享公用模式的全部或部分，包括公用数据对象、键值对以及其他标识符和结构，从而将相关复制品耦合在一起。共享模式允许同一查询用于复制品两者。此外，每个复制品能够被本地修改并在随后通过被称为“同步”的过程用其他相关复制品来更新。通过同步，一个复制品中数据对象的改变就被记录并作为同步更新传播到其他相关复制品，其中对相关复制品的相应数据对象执行相同的改变。

[0004] 然而，当在各分开复制品中的相应数据对象被并发地修改时会出现冲突。一般而言，冲突表示在一个复制品 A 处对一数据对象进行修改，随后在 A 处的改变信息经通信链接传播至 B 之前就在另一复制品 B 处对相应的数据对象进行修改。例如，某产品的价格可以在一个复制品处被改变为 \$X，而几乎与此同时（例如，在与其他复制品同步每个改变之前），相同产品的价格可以在另一复制品处被改变为 \$Y，由此就在两个复制品之间对于该产品的价格产生了冲突。为了将复制品恢复到一致状态，冲突可被检测并在随后被解决。

[0005] 存在不同的策略来检测冲突。例如，可以跟踪一复制品已被修改的时间，并在同步期间将其传播给其他复制品。一旦接收到同步更新，每个接收复制品就能够通过比较远程复制品处数据对象的改变时间和另一相应数据对象（无论本地或远程）的改变时间来检测冲突。

[0006] 在检测到冲突之后，分布式数据库系统能够工作以解决冲突。同样存在不同的策略来解决这些冲突。例如，冲突可以在算法上解决，通过用户交互解决等。然而，由于复制数据库系统的分布式特性和同步的异步特性，因而会在不同复制品处同时检测到相应数据对象内的竞争冲突。竞争冲突的检测因此能够触发竞争冲突解决并将其传播至整个分布式数据库，藉此引入新的冲突。如此，现有的数据库服务器易于在冲突解决期间停止数据库处理，从而避免这一竞争冲突解决。此外，竞争冲突解决尤其需要挑战“非齐次 (non-idempotent)”冲突（即，不同的复制品会不同地解决冲突的情况或者解决会激起遍布该分布式数据库的显著改变的时刻）。因此，如何在不停止其他数据库处理的情况下处理分布式数据库内不同复制品处的冲突解决的问题就变得相关。

[0007] 概述

[0008] 本文描述的并声明的各实现通过响应于对一个或多个同步冲突的检测而和每个复制品的同步过程并发地运行一个或多个解决过程来解决上述问题。每个解决过程用于

解决有关数据对象的冲突而同步过程则继续处理其他同步更新。在冲突解决已被确定之后，针对本地复制品的当前条件测试冲突解决。如果冲突解决不陈旧（例如，在本地解决过程期间没有外部冲突解决被接收），则将其应用于本地复制品并传递回到同步过程以供传播至远程复制品。否则，如果冲突解决是陈旧的（例如，一个外部冲突解决已从另一复制品被接收），则本地冲突解决被丢弃。

[0009] 在一些实现中，提供作为计算机程序产品的制品。计算机程序产品的一个实现提供可由计算机系统读取并编码计算机程序的计算机程序存储介质。由计算系统在载波内具体化并编码计算机程序的计算机数据信号可提供计算机程序产品的另一个实现。在此还描述和列举其它实现。

[0010] 提供本概述以便以简化形式介绍将在以下详细描述中进一步描述的一些概念。本概述并不旨在标识要求保护的主题的关键特征或本质特征，也不旨在用于帮助确定要求保护的主题的范围。

附图说明

- [0011] 图 1 示出了描述一示例性分布式冲突解决的分布式数据库系统。
- [0012] 图 2 示出了描述另一示例性分布式冲突解决的分布式数据库系统。
- [0013] 图 3 示出了用于分布式数据库系统的示例性同步和解决过程。
- [0014] 图 4 示出了可用于实现所描述技术的示例性系统。

具体实施方式

[0015] 图 1 示出了描述示例性分布式冲突解决的分布式数据库系统 100。分布式数据库系统 100 可以具有分布式相关数据库、文件系统、数据日志或其他经组织数据收集的形式。第一数据库服务器 102、第二数据库服务器 104 和第三数据库服务器 106 经由诸如因特网或局域网 (LAN) 等的网络 108 通信连接。每个数据库服务器都可以访问分布式数据库的各别复制品。每个复制品与分布式数据库系统 100 中的其他复制品共享一完整或部分模式。此外，各复制品由同步框架耦合，在其中每个复制品经由同步更新将其改变传播至其他相关复制品，而后者随后可以用所传播的改变更新其自身的数据对象。

[0016] 同步指的是用另一复制品的改变的数据对象的数据来更新一个复制品的数据对象，使得第一数据对象反映出对该改变的数据对象所做出的改变。在一个实现中，例如本地复制品的同步作为背景线程或过程来执行。如果数据对象的远程副本已被改变并且数据对象的本地副本尚未被更新以反映该改变，则同步实行对数据对象本地副本的改变。此外，同步将本地改变作为同步更新持续传播至其他相关复制品，并且处理从远程复制品接收的同步更新以更新本地数据。使用这些特征，一个复制品中的改变就可以被同步到分布式数据库系统中的所有其他相关复制品内。

[0017] 然而，如果并发地检测到两个不同复制品内相应数据对象的改变，则会导致冲突。例如，在图 1 中，与数据库服务器 102 相关联的复制品的数据对象 110 用新价格更新，得到更新的数据对象 112（参见包含经改变价格的阴影单元格）。改变操作被传播至其他相关复制品（参见分支到复制品 104 和 106 的虚线箭头 114）。然而，在来自与数据库服务器 104 相关联的复制品的改变进行传播的同时，与数据库服务器 104 相关联的复制

品的相应数据对象 116 用新价格更新，得到更新的数据对象 118（参见包含经改变价格的阴影单元格）。结果在数据库服务器 102 处，就会检测到与数据库服务器 102 和 104 相关联的复制品的相应数据对象之间的冲突。

[0018] 可以使用不同的策略来检测这些冲突。在一个实现中，分布式数据库系统利用分布式时间戳机制并使用同步参数来检测冲突。同步参数可以显式地指代向量时间对、实时值、版本向量、或者可用于跟踪彼此相关的更新事件的其他值。

[0019] 在一个实现中，每个复制品维持本地时间戳，后者只要在该复制品处的数据状态改变时就递增，而数据状态的改变或者由于本地更新或者由于与对等复制品的同步。对于每个数据对象，该复制品还维持两个同步参数：(1) 修改参数 m ，(为每个本地已知的复制品 R) 指示 R 处数据对象最后改变的 R 处时间戳；以及 (2) 同步参数 s ，(为每个本地已知的复制品 R) 指示 R 处数据对象最后同步的 R 处时间戳(例如，由远程同步更新产生)。在一个实现中，参数 m 和 s 被表示为带有对应于每个已知复制品的同步参数的时间戳向量，虽然也可以利用其他参数类型。通俗地，修改参数 m 跟踪数据对象的本地版本，而同步时间参数 s 跟踪数据对象相对于其他复制品有多当前。同步过程比较修改时间参数 m_A 和同步时间参数 s_A (与远程复制品 A 的数据对象的同步更新相关联)的组合与修改时间参数 m_B 和同步时间参数 s_B (与本地复制品 B 的对应本地数据对象相关联)的组合，以确定是否更新本地数据对象。

[0020] 例如，当考虑复制品 B 中的本地数据对象是否需要相对于复制品 A 被更新时， $m_A \leq s_B$ 测试复制品 B 中的本地数据对象是否由复制品 A 的对应数据对象中的改变同步。因此，如果 $m_A \leq s_B$ ，复制品 B 无需被更新，因为复制品 B 已经与从复制品 A 获取的最近修改相同步。相反地，如果 $m_A > s_B$ ，则 $m_B \leq s_A$ 测试复制品 A 是否相对于来自复制品 B 的最后修改而被同步。因此，如果 $m_A > s_B$ 且 $m_B \leq s_A$ ，则复制品 A 具有尚未与复制品 B 同步的更近修改，所以复制品 B 需要来自复制品 A 的同步更新来更新。然而，如果 $m_A > s_B$ 且 $m_B > s_A$ ，则复制品 A 与 B 两者都已经用并发修改来修改并且存在冲突。应该理解也可以利用其他冲突检测模式。

[0021] 在检测到冲突之后，数据库服务器通常会尝试使用各种冲突解决技术中的一种来解决冲突。在一个实现中，可以向用户呈现冲突并且向其询问以选择期望的结果，随后该结果可以传播遍及分布式数据库系统 100。在其它实现中，冲突可在算法上被解决(例如，基于时间向量选择最近改变，选择最高改变值等)。也可以利用其他冲突解决技术。

[0022] 数据库服务器 102 检测冲突并用于解决该冲突。在一个实现中，在数据库服务器 102 上执行的行进中的同步过程产生冲突解决线程并在随后继续与接收自其他复制品的其他同步更新相同步。这些更新可以包括针对已在远程复制品处生成的相应冲突的冲突解决。在确定了冲突解决之后，数据库服务器 102 就确定该解决是否陈旧。陈旧解决指的是因为来自远程复制品的介入冲突同步更新已经在本地复制品的冲突解决处理期间应用于相同数据对象而不再有效的冲突解决。因此在此情况下，冲突已由远程数据库服务器解决，或者数据对象已通过其他途径改变，并且由解决过程确定的本地冲突解决是陈旧的。如果数据库服务器 102 确定本地冲突解决不陈旧，则该解决就被本地应用并传播至分布式数据库系统中的其他相关复制品。

[0023] 图 2 示出了描述另一示例性分布式冲突解决的分布式数据库系统 200。第一数据库服务器 202、第二数据库服务器 204 和第三数据库服务器 206 经由诸如因特网或局域网 (LAN) 等的网络 208 通信连接。每个数据库服务器都可以访问分布式数据库的各别复制品。每个复制品与分布式数据库系统 200 中的其他复制品共享一完整或部分模式。此外，各复制品由同步框架耦合，在其中每个复制品将其改变传播至其他相关复制品，而后者随后可以用传播的改变更新其自身数据。

[0024] 如同图 1 的示例一样，会产生冲突。例如，在图 2 中，与数据库服务器 202 相关联的复制品的数据对象 210 用新价格更新，得到更新的数据对象 212(参见包含经改变价格的阴影单元格)。改变操作被传播至其他相关复制品 (参见将改变操作传播至复制品 206 的虚线箭头 214，应该理解该改变操作也可传播至复制品 204)。然而，在来自与数据库服务器 204 相关联的复制品的改变进行传播的同时，与数据库服务器 204 相关联的复制品的数据对象 216 用新价格更新，得到更新的数据对象 218(参见包含经改变价格的阴影单元格)。改变操作被传播至其他相关复制品 (参见将改变操作传播至复制品 206 的虚线箭头 215，应该理解该改变操作也可传播至复制品 202)。结果在数据库服务器 206 处，就会检测到与数据库服务器 202 和 204 相关联的复制品的相应数据对象之间的冲突。

[0025] 数据库服务器 206 检测冲突并用于解决该冲突。在一个实现中，在数据库服务器 206 上执行的行进中的同步过程产生冲突解决线程并在随后继续与接收自其他复制品的其他更新相同步。在确定了冲突解决之后，数据库服务器 206 就确定该解决是否陈旧。陈旧解决指的是因为来自远程复制品或者本地修改的介入冲突同步更新已经在本地复制品的冲突解决处理期间应用于相同数据对象而不再有效的冲突解决。因此在此情况下，冲突已由远程数据库服务器解决，或者数据对象已通过其他途径改变，并且由解决过程确定的本地冲突解决是陈旧的。如果数据库服务器 206 确定本地冲突解决不陈旧，则该解决就被本地应用并传播至分布式数据库系统中的其他相关复制品。

[0026] 图 3 示出了用于分布式数据库系统的示例性同步和解决过程 300。同步过程通常在每台数据库服务器上执行，接收来自远程复制品的远程同步更新并将本地同步更新传播至那些远程复制品。每个同步更新包括同步参数，诸如版本向量、向量时间对或其他参数。例如，同步参数可以具有包含本地已知的有关在每个已知远程复制品处数据对象的同步或修改状态的信息的向量的形式。

[0027] 接收操作 302 接收来自更新队列的同步更新。更新队列包括从远程复制品接收的同步更新以及要从本地复制品传播的同步更新 (例如，本地冲突解决、本地同步或其他本地修改)。如果该同步更新由决定操作 304 判定为本地 (例如，原始本地修改、本地确定的冲突解决、或者由远程同步更新产生的本地更新)，则传播操作 306 将该同步更新发送至其他复制品并且处理返回到接收操作 302 以接收更新队列中的下一个同步更新。该同步更新由决定操作 304 判定为来自远程复制品 (例如，远程同步更新或者远程确定的冲突解决)，则锁定操作 308 锁定本地复制品中的数据对象以避免远程同步更新和本地冲突解决应用之间的竞争条件。审查操作 309 随后检查同步更新的同步参数以确定冲突是否存在，诸如使用前述修改参 (m) 数和同步参数 (s) 方法。如果未在检测操作 310 中检测到冲突，则更新操作 312 或者将来自同步更新的改变应用于本地复制品或者什么也不做 (例如，如果本地复制品已经被同步)。

[0028] 在一个实现中，可以应用来自更新的两类不同改变，虽然在可选实现中可以只应用一类或两类以上。在第一类改变中，更新导致本地复制品中一状态（例如，数据）改变。例如，同步更新可以指示本地复制品中数据对象的值。作为响应，更新操作 312 改变数据对象的值并更新与该数据对象相关联的同步参数。在第二类改变中，更新不导致本地复制品中的状态改变。例如，在同步更新中指示的值与本地数据对象中已有值相同。作为响应，更新操作 312 只需更新与该数据对象相关联的同步参数而同步更新的剩余部分（例如，值改变）可被丢弃。

[0029] 在更新操作 312 之后，解锁操作 313 解除该数据对象的锁定，而排队操作 315 随后将同步更新作为本地更新传递给更新队列，该更新于是能够在同步过程的下一阶段被正常处理。作为替换，同步更新可以被直接传递给传播操作 306。处理随后行进至传播操作 306 以在其返回到接收操作 302 来接收来自更新队列的另一同步更新之前通知改变的其他远程复制品。

[0030] 如果检测操作 310 检测到冲突，则快照操作 314 就通过记录有关同步更新和冲突中本地数据对象的信息来捕捉该冲突的快照，上述信息诸如可以是本地复制品的当前时间戳、本地数据对象的状态（例如，值）、远程数据对象的状态、本地同步参数、以及远程同步参数。新的过程操作 316 初始（例如，产生）由该快照伴随的独立解决过程（例如，新线程）以确定该冲突的解决。解决过程与关联于该冲突的数据对象相关联，从而可以为不同的数据对象同时运行多个解决过程。在初始解决过程之后，同步过程执行解除数据对象锁定的解锁操作 317，并在随后返回到接收操作 302 同时解决过程继续处理该冲突。当一个或多个过程并发操作时，每个过程能够与多任务或多线程环境中一个或多个处理器上的其他过程协同执行指令。这一并发处理允许同步过程用同步更新继续更新本地复制品并传播其他改变，同时解决过程处理该冲突。

[0031] 在解决过程中，接收操作 318 接收来自同步过程的快照。解决操作 320 解决该冲突，诸如通过应用冲突解决算法或者要求用户从两个数据状态（本地或远程）中选择一个。在解决该冲突之后，锁定操作 322 锁定本地复制品中的数据对象以避免远程同步更新和本地冲突解决应用之间的竞争条件。

[0032] 在向本地复制品应用经解决的更新之前，解决过程检测该冲突解决是否陈旧。陈旧暗示在该解决被处理期间，接收另一远程同步更新并将其应用于同一本地数据对象。因此，陈旧测试操作 324 记录解决操作 320 之后存在的条件并将从同步过程的快照中接收到的本地时间戳与关联于本地数据对象的当前同步参数（例如，在解决操作 320 之后捕捉的）相比较。如果同步参数比来自快照的时间戳更近，则本地冲突解决是陈旧的。因此，解锁操作 326 就解除复制品中数据对象的锁定，而删除操作 328 删除该解决。

[0033] 相反地，如果陈旧测试操作 324 确定本地冲突解决不陈旧，更新操作 330 就把该冲突解决应用于本地复制品，包括更新本地数据对象的数据并更新本地数据对象的同步参数以反映该冲突解决的应用。其后，解锁操作 332 解除本地复制品中数据对象的锁定。排队操作 334 随后将冲突解决作为本地更新传递给更新队列，该更新能够在同步过程中被正常处理。作为替换，冲突解决可以被直接传递给传播操作 306。

[0034] 终止操作 336 终止该解决过程。如上所述，可以为由关联于本地复制品执行的单个同步过程初始的不同冲突并发执行多个解决过程。

[0035] 还应理解当在同步过程中接收到针对一数据对象的介入同步更新并对其进行处理时，在对该同一数据对象执行解决过程期间，可以检测到新的潜在冲突。在此情况下，如果同步更新包括与当前本地状态相兼容（例如，相等）的远程数据状态（例如，表示无冲突），就更新本地数据对象，这包括更新同步参数，并且相关联的解决过程可以被中断并丢弃。相反地，如果同步更新包括与当前本地状态不兼容（例如，不相等）的远程数据状态（例如，表示冲突），就丢弃该同步更新以支持与该数据对象相关联的进行中的冲突解决。

[0036] 用于实现本发明的图 4 的示例性硬件和操作环境包括具有游戏控制台或计算机 20 形式的通用计算设备，包括处理单元 21、系统存储器 22、将包括系统存储器在内的各种系统组件操作性地耦合至处理单元 21 的系统总线 23。可以只存在一个或者存在一个以上处理单元 21，使得计算机 20 的处理器包括单中央处理单元 (CPU)，或者多处理单元，通常称为并行处理环境。计算机 20 可以是常规计算机，分布式计算机，或者任何其他类型的计算机，本发明在此不受限制。

[0037] 系统总线 23 可以是几种类型的总线结构中的任何一种，包括存储器总线或存储控制器、外围总线、切换结构、点对点连接、以及使用各种总线体系结构中的任一种的局部总线。系统存储器可被简单称为存储器，并且包括只读存储器 (ROM) 24 和随机存取存储器 (RAM) 25。基本输入 / 输出系统 (BIOS) 26 存储在 ROM 20 中，包含帮助在诸如启动期间在计算机 24 内元件之间传递信息的基本例程。计算机 20 还包括用于对硬盘（未示出）进行读写的硬盘驱动器 27、用于对可移动磁盘 29 进行读写的磁盘驱动器 28 以及用于对可移动光盘 31，如 CD-ROM 或其它光介质进行读写的光盘驱动器 30。

[0038] 硬盘驱动器 27、磁盘驱动器 28 和光盘驱动器 30 分别由硬盘驱动器接口 32、磁盘驱动器接口 33 和光盘驱动器接口 34 连接到系统总线 23。驱动器及其关联的计算机可读媒质为计算机 20 提供了计算机可读指令、数据结构、程序模块和其它数据的非易失性存储。本领域的技术人员可以理解，在示例操作环境中也可使用可储存可由计算机访问的数据的任何类型的计算机可读媒质，如磁带盒、闪存卡、数字视频盘、随机存取存储器 (RAM)、只读存储器 (ROM) 等等。

[0039] 若干程序模块能存储在硬盘、磁盘 29、光盘 31、ROM 24 或 RAM 25 中，包括操作系统 35、一个或多个应用程序 36、其它程序模块 37 和程序数据 38。用户可通过诸如键盘 40 和定点设备 42 的输入设备向个人计算机 20 中输入命令和信息。其它输入设备（未示出）可以包括麦克风、操纵杆、游戏手柄、圆盘式卫星天线、扫描仪等。这些和其它输入设备通常通过耦合到系统总线的串行端口接口 46 连接到处理单元 21，但也可通过其它接口连接，如并行端口、游戏端口和通用串行总线 (USB)。监视器 47 或其它类型的显示设备也经由接口，诸如视频接口 48 连接至系统总线 23。除监视器之外，计算机通常包括其它外围输出设备（未示出），如扬声器或打印机。

[0040] 计算机 20 可使用至一个或多个远程计算机，诸如远程计算机 49 的逻辑连接在网络化环境中操作。这些逻辑连接由耦合至或者作为计算机 20 一部分的通信设备来实现，本发明不限于特定类型的通信设备。远程计算机 49 可以是另一计算机、服务器、路由器、网络 PC、客户机、对等设备或其它公共网络节点，并且通常包括许多或所有以上相对于计算机 20 所描述的元件，尽管在图 4 中仅示出了存储器存储设备 50。图 4 中画出

的逻辑连接包括局域网 (LAN) 51 和广域网 (WAN) 52。这样的网络环境常见于办公室网络、企业范围计算机网络、内联网和因特网，并且可以是所有类型的网络。

[0041] 当在 LAN 网络环境中使用时，计算机 20 通过作为一种通信设备类型的网络接口或适配器 53 连接到局域网 51。当在 WAN 联网环境中使用时，计算机 20 通常包括调制解调器 54、网络适配器、一种类型的通信设备，或者用于经广域网 52 建立通信的任何其他类型的通信设备。或为内置或为外置的调制解调器 54 经由串行端口接口 46 连接到系统总线 23。在网络化环境中，相对于个人计算机 20 所描述的程序模块或其部分可以存储在远程存储器存储设备中。可以理解，所示的网络连接是示例性的，且可以使用在计算机之间建立通信链路的其它手段和通信设备。

[0042] 在示例性实现中，同步模块、解决模块以及其它模块可由存储在存储器 22 和 / 或存储设备 29 或 31 中并由处理单元 21 处理的指令来具体化。复制品、快照、冲突解决、同步更新和其他数据可以存储在存储器 22 内和 / 或作为永久性数据存储的存储设备 29 或 31 内。

[0043] 此处所描述的技术在一个或多个系统中被实现为逻辑操作和 / 或模块。逻辑操作可被实现为在一个或多个计算机系统中执行的处理器实现的步骤的序列，以及在一个或多个计算机系统中的互连机器或电路模块。同样地，各组件模块的描述可以按照由模块执行或影响的操作来提供。所得的实现的选择取决于实现所描述的技术的下层系统的性能需求。从而，组成此处所述的技术的实施例的逻辑操作被不同地表示为操作、步骤、对象或模块。此外，应该理解，除非以其他方式明确地提出权利要求或权利要求语言固有地需要一特定的次序，否则逻辑操作可以任何次序来执行。

[0044] 以上说明、示例和数据提供了对本发明的各示例性实施例的结构和使用的全面描述。尽管以上带着一定程度的特殊性或对一个或多个单独实施例的参考描述了本发明的各实施例，但是本领域的技术人员能够对所公开的实施例做出多种改变而不背离本发明的精神或范围。更具体地，应该理解，可以独立于个人计算机采用所描述的技术。因此可考虑其他实施例。其目的是以上描述中所包含以及在附图中所示出的所有一切应当解释为仅仅是例示特定实施例而非限制。可以对细节或结构做出更改而不背离如以下权利要求书中定义的本发明的基本元素。

[0045] 虽然已经用具体到结构功能部件和 / 或方法技艺的语言对主题进行了描述，但是应该理解，在所附权利要求中定义的该主题不必限于上述具体的功能部件或动作。相反，上述具体特征和动作是作为实现权利要求主题的示例形式公开的。

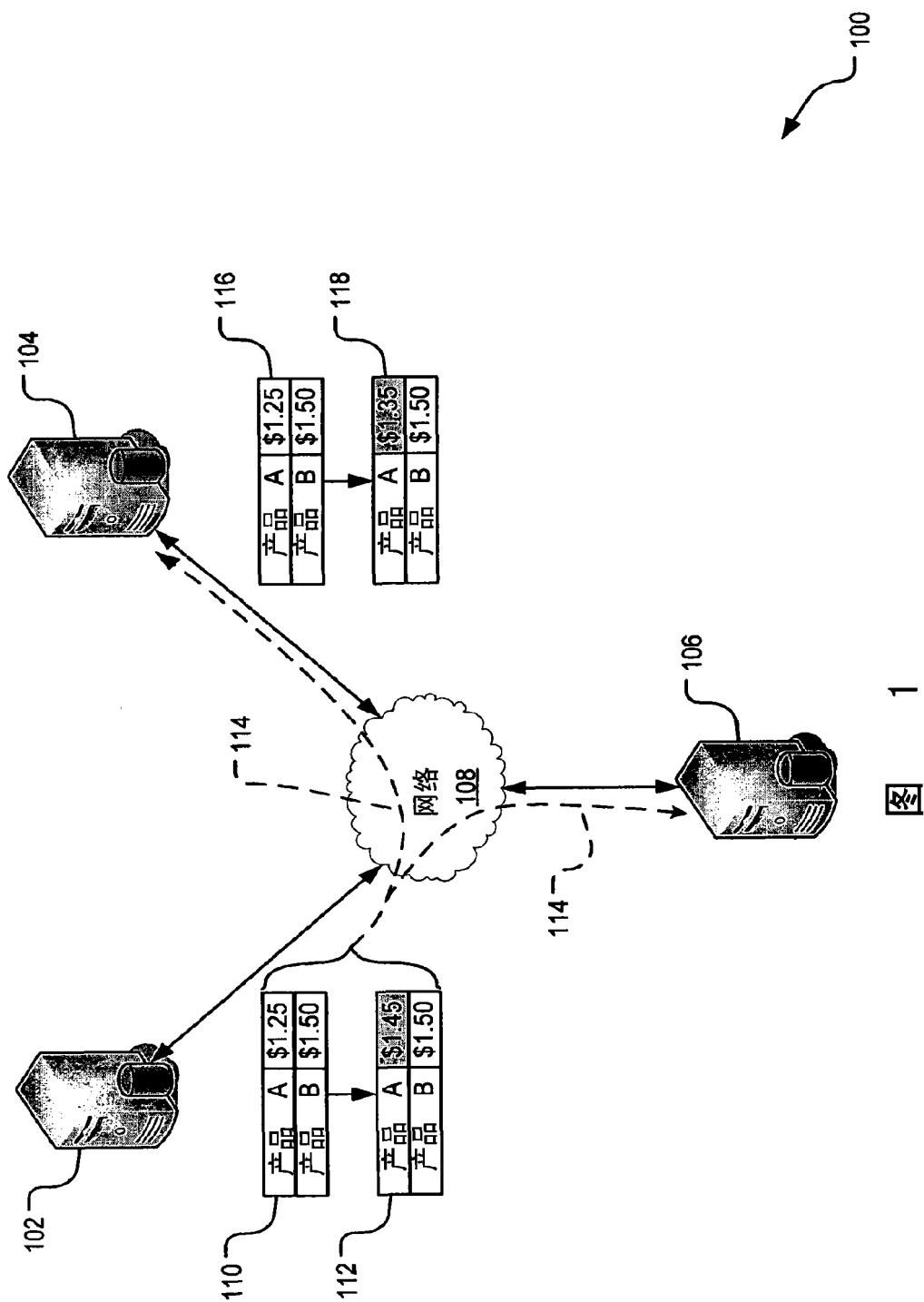


图 1

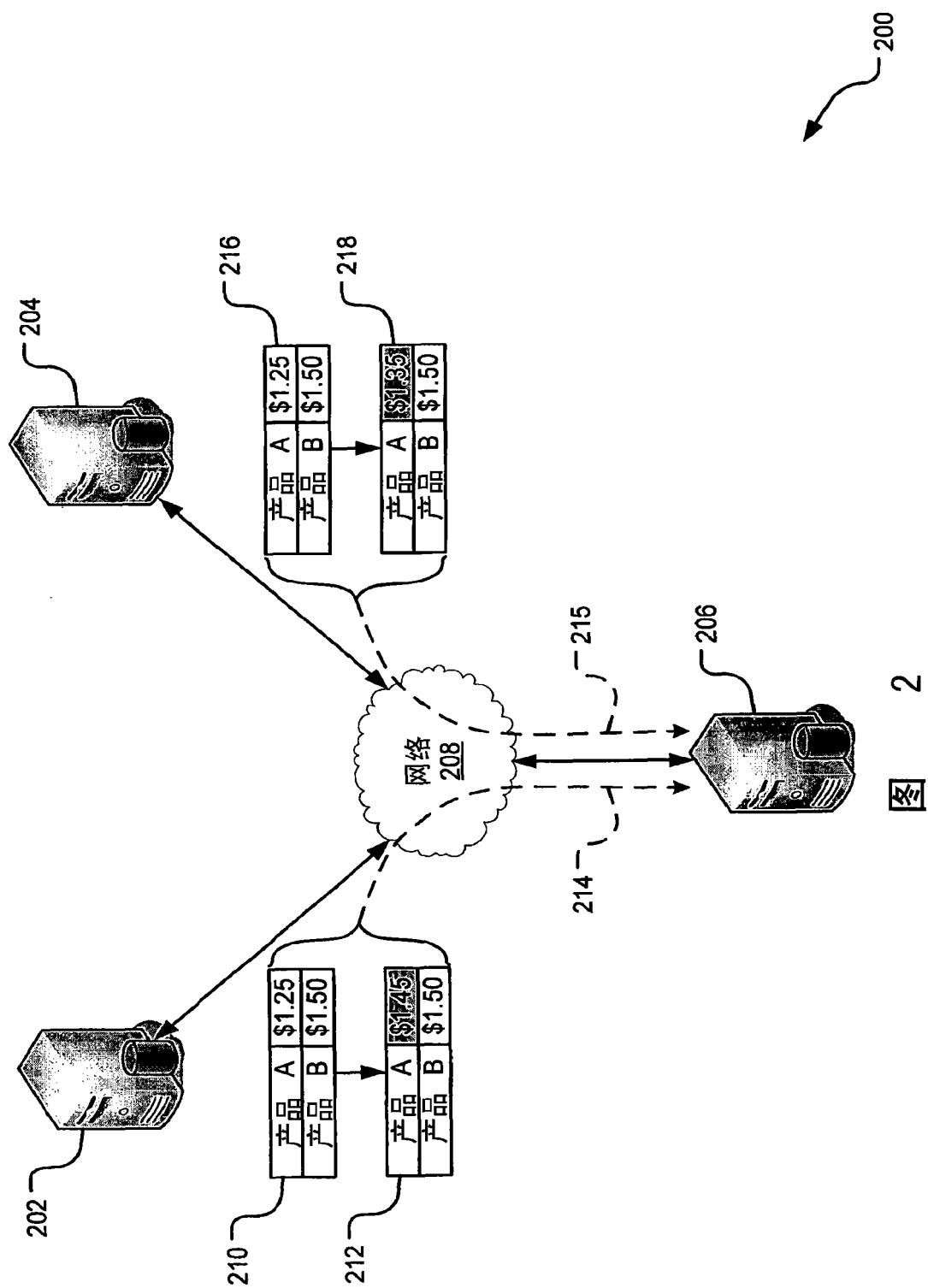


图 2

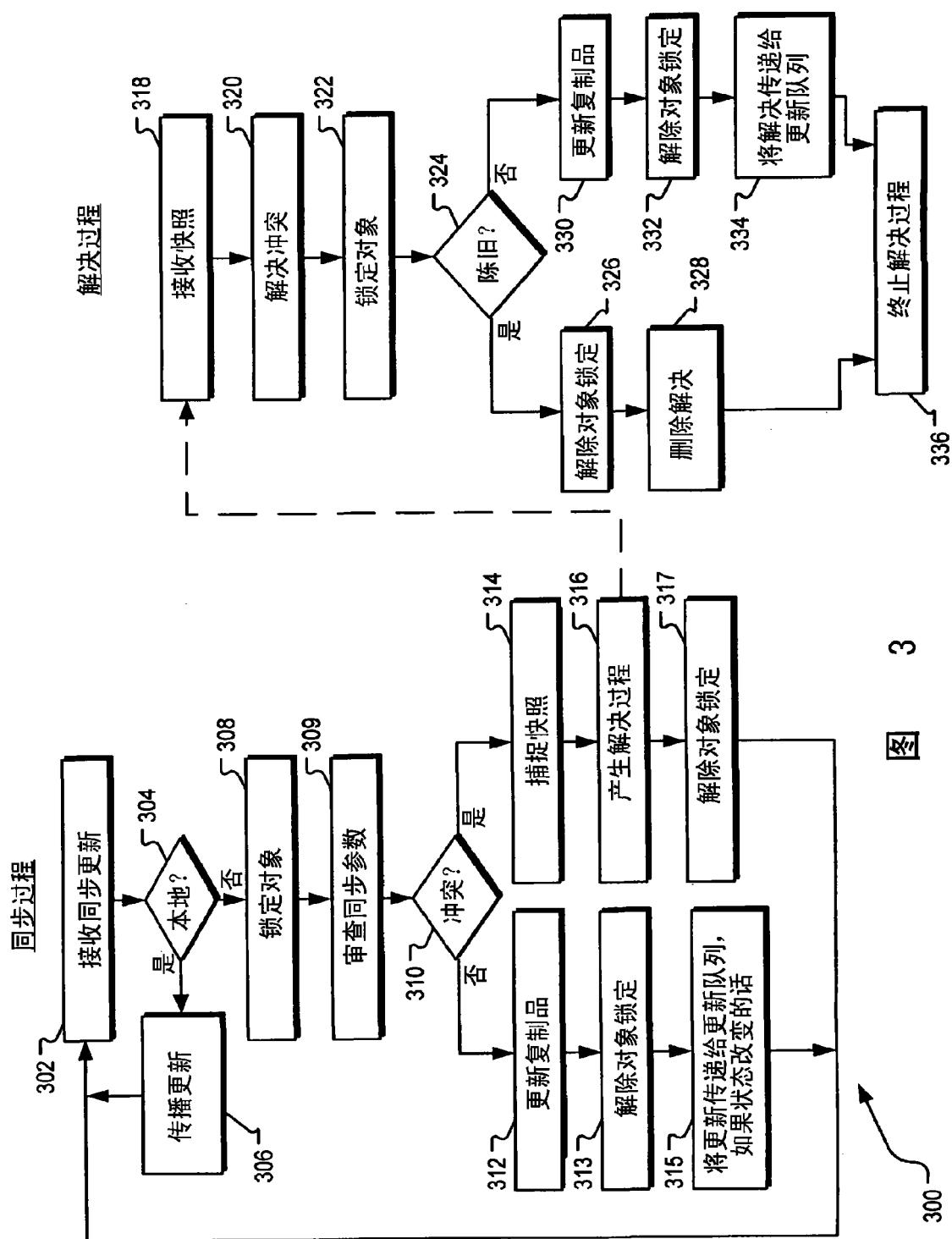


图 3

