(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 108139927 A
(43)申请公布日 2018.06.08

(21)申请号 201680057373.X

(22)申请日 2016.09.27

(30)优先权数据
14/873,070 2015.10.01 US

(85)PCT国际申请进入国家阶段日
2018.04.02

(86)PCT国际申请的申请数据
PCT/CN2016/100413 2016.09.27

(87)PCT国际申请的公布数据
WO2017/054711 EN 2017.04.06

(71)申请人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72)发明人 陈萌萌 阿尼凯特·阿德纳 帕布朗·阿潘德朗

(51)Int.Cl.
G06F 9/46(2006.01)
G06F 17/00(2006.01)

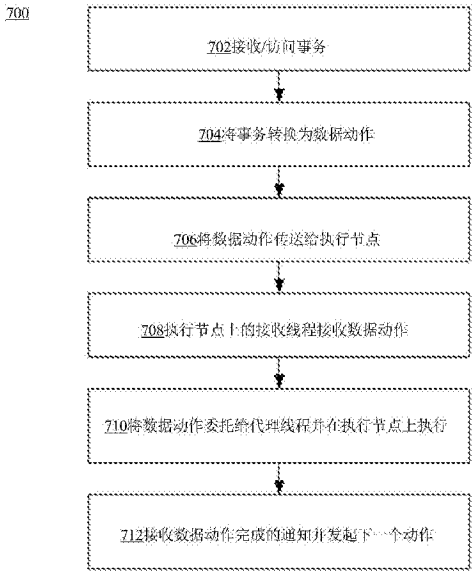
权利要求书3页 说明书11页 附图6页

(54)发明名称

联机事务处理系统中事务的基于动作的路由

(57)摘要

节点访问对联机事务处理系统中的数据库执行事务的请求。确定数据库中事务待作用的数据集合。然后根据动作的数据依赖关系将事务分离为多个动作;针对事务作用的每个数据集合创建一个动作。将这些动作传送到存储相应动作依赖的数据的节点。然后在这些动作路由到的节点上执行这些动作。



CN 108139927 A

1.一种由联机事务处理(online transaction processing,简称OLTP)系统中的节点执行的方法,其特征在于,所述节点包括处理器和存储器,所述方法由所述处理器执行并包括:

在所述节点处接收来自客户端的用于对所述OLTP系统中的数据库执行事务的请求,所述事务包括标识所述事务待作用的多个数据集合的信息,所述信息至少标识存储在所述OLTP系统的第一节点上的第一数据集合和存储在所述OLTP系统的第二节点上的第二数据集合;

定义作为所述事务一部分的动作,所定义的动作取决于所述OLTP系统中哪些节点存储了所述事务待作用的数据,所述动作包括与所述第一数据集合相关联的第一动作,还包括与所述第二数据集合相关联的第二动作;

将所述第一动作传送给所述第一节点,并将所述第二动作传送给所述第二节点,所述第一节点可操作用于对所述第一数据集合执行所述第一动作,所述第二节点可操作用于对所述第二数据集合执行所述第二动作。

2.根据权利要求1所述的方法,其特征在于,所述数据库在逻辑上和物理上划分成分区,所述分区中的每一个都包括一行或多行数据,所述数据库中的一个数据集合包括分区的一行或多行,所述方法还包括:

识别所述事务待作用的所述数据库中的分区的分区标识(identifier,简称ID);

访问所述分区ID到存储所述分区的节点的映射;

基于所述分区ID和所述存储所述分区的节点将所述事务分离为包括所述第一动作和所述第二动作的所述动作。

3.根据权利要求1所述的方法,其特征在于,所述传送所述第一动作和所述传送所述第二动作进一步包括:

向第一线程发送针对所述第一动作的动作特定第一消息,所述第一线程在所述第一节点上执行时将所述第一动作委托给第二线程,所述第二线程在所述第一节点上执行时对所述第一数据集合执行所述第一动作;

向第三线程发送针对所述第二动作的动作特定第二消息,所述第三线程在所述第二节点上执行时将所述第二动作委托给第四线程,所述第四线程在所述第二节点上执行时对所述第二数据集合执行所述第二动作。

4.根据权利要求3所述的方法,其特征在于,所述第二线程将所述第一动作入队,并在所述第一动作入队时向所述第一线程发送消息;所述第四线程将所述第二动作入队,并在所述第二动作入队时向所述第三线程发送消息。

5.根据权利要求3所述的方法,其特征在于,所述第二线程在执行所述第一动作之前调用所述第一数据集合的锁定,并在所述第一动作完成时向所述第一线程发送消息,所述第四线程在执行所述第二动作之前调用所述第二数据集合的锁定,并在所述第二动作完成时向所述第三线程发送消息。

6.根据权利要求1所述的方法,其特征在于,还包括:从所述第一节点和第二节点接收所述第一动作和第二动作完成的通知。

7.根据权利要求6所述的方法,其特征在于,还包括:响应于接收到所述通知,指示所述第一节点和第二节点分别提交所述第一动作和第二动作。

8.根据权利要求6所述的方法，其特征在于，还包括：响应于接收到所述通知，将所述事务的第三动作传送给存储有所述第三动作待作用的数据集合的节点。

9.一种联机事务处理(online transaction processing，简称OLTP)系统中的节点，其特征在于，所述节点包括：

处理电路；

与所述处理电路耦合的存储器，其中存储着指令，当所述指令执行时，使得所述节点执行方法，所述方法包括：

在所述节点处接收数据库上待执行的事务，所述数据库在物理上划分为存储在所述OLTP系统中多个节点上的数据集合；

基于所述事务标识的分区号，确定所述事务待作用于哪个数据集合，其中，所述确定至少标识存储在所述多个节点中的第一节点上的集合的第一集合，以及存储在所述多个节点中的第二节点上的集合的第二集合；

根据所述数据集合中所述事务待作用的具体数据集合以及所述OLTP系统中存储有所述事务待作用的数据集合的具体节点，将所述事务分离为动作，所述动作包括至少第一动作和第二动作，所述第一动作映射到所述第一数据集合，所述第二动作映射到所述第二数据集合；

将所述第一动作传送给所述第一节点，并将所述第二动作传送给所述第二节点，所述第一节点和第二节点分别可操作用于执行所述第一动作和第二动作；

从所述第一节点和第二节点接收所述第一动作和第二动作完成的通知；

响应于所述通知，发起在所述第一动作和第二动作之后的所述事务的下一个动作。

10.根据权利要求9所述的节点，其特征在于，所述方法还包括：

访问将所述事务标识的分区号映射到存储数据集合的节点的信息；

基于与所述数据集合相关联的分区号，将所述事务分离为包括所述第一动作和所述第二动作的所述动作。

11.根据权利要求9所述的节点，其特征在于，所述传送所述第一动作和所述传送所述第二动作包括：

向在所述第一节点上执行的第一线程发送包括描述所述第一动作的信息的第一动作特定消息，以将所述第一动作从所述第一线程委托给在所述第一节点上执行的第二线程，以对所述第一数据集合执行所述第一动作；

向在所述第二节点上执行的第三线程发送包括描述所述第二动作的信息的第二动作特定消息，以将所述第二动作从所述第三线程委托给在所述第二节点上执行的第四线程，以对所述第二数据集合执行所述第二动作。

12.根据权利要求11所述的节点，其特征在于，所述第二线程将所述第一动作入队，在所述第一动作入队时向所述第一线程发送消息，在执行所述第一动作之前调用所述第一数据集合的锁定，并在所述第一动作完成时向所述第一线程发送消息；所述第四线程将所述第二动作入队，在所述第二动作入队时向所述第三线程发送消息，在执行所述第二动作之前调用所述第二数据集合的锁定，并在所述第二动作完成时向所述第二线程发送消息。

13.根据权利要求9所述的节点，其特征在于，所述下一个动作包括：指示所述第一节点和第二节点分别提交所述第一动作和第二动作。

14.根据权利要求9所述的节点，其特征在于，所述下一个动作包括：将所述事务的第三动作路由到所述多个节点中存储有所述第三动作待作用的数据集合的节点。

15.一种具有计算机可执行指令的非瞬时性计算机可读存储介质，其特征在于，当所述指令执行时，使得联机事务处理(online transaction processing，简称OLTP)系统中的路由节点执行方法，所述方法包括：

访问所述OLTP系统中的数据库上待执行的事务，所述数据库在物理上和逻辑上划分为存储在多个执行节点上的多个数据集合；

根据所述数据集合中包括所述事务待作用的数据的具体数据集合，并根据所述执行节点中存储有所述事务待作用的数据的具体节点，将所述事务分离为多个动作，其中，所述多个动作中的每个动作都与存储在所述执行节点中相应一个节点上的数据集合中相应一个数据集合相关联；

对于所述多个动作中的每个动作，将动作特定消息发送到存储所述动作的数据集合的所述执行节点上的接收线程，其中，所述接收线程将所述动作委托给所述执行节点上的执行线程，当所述执行线程执行时，执行所述动作；

响应于接收到指示所述多个动作中的每个动作完成的通知，发起与所述事务相关联的另一动作。

16.根据权利要求15所述的计算机可读存储介质，其特征在于，所述方法还包括：

识别所述事务待作用的数据的分区的分区标识(identifier，简称ID)；

访问所述分区ID到存储所述分区的执行节点的映射；

基于所述分区ID和存储所述事务待作用的数据的分区的所述执行节点，将所述事务分离为多个动作。

17.根据权利要求15所述的计算机可读存储介质，其特征在于，所述执行线程将所述动作入队，并在所述动作入队时向所述接收线程发送消息。

18.根据权利要求17所述的计算机可读存储介质，其特征在于，所述执行线程在执行所述动作之前调用所述数据集合的锁定，并在所述动作完成时向所述接收线程发送消息。

19.根据权利要求15所述的计算机可读存储介质，其特征在于，所述发起包括：指示所述执行节点提交所述动作。

20.根据权利要求15所述的计算机可读存储介质，其特征在于，所述发起包括：将所述事务的另一动作路由到存储所述另一动作待作用的数据集合的执行节点。

联机事务处理系统中事务的基于动作的路由

[0001] 相关申请案交叉申请

[0002] 本申请要求于2015年10月1日递交的发明名称为“联机事务处理系统中事务的基于动作的路由”的第14/873,070号美国申请案的在先申请优先权,该在先申请的内容以引用的方式并入本文。

背景技术

[0003] 联机事务处理(online transaction processing,简称OLTP)用于促进事务数据(例如,事务相关数据)的存储、检索和维护。OLTP用于高度依赖高效处理大量客户端事务的行业(如银行、航空业和零售业)。支持OLTP的数据库系统通常分布在多个服务器上,以避免单点故障并分散数据量和流量。

[0004] 随着数据量的增加和新类型事务的引入,与高吞吐量OTLP数据库和系统相关联的需求正在急剧增长。由于商业模式的复杂性增加,传统较小的事务正在让位于更大更复杂的事务。

[0005] 一些当前OLTP系统尝试通过采用事务间并行性来提高性能。这些类型的系统调度多个工作线程同时执行,每个线程自己运行一个完整事务。但是,这些类型的系统可能存在一些问题,例如指令数据位置较差;即,一台服务器上执行的线程可能需要作用于驻留在其他服务器上的数据。为了获得该数据,线程将数据库查询以例如结构化查询语言(structured query language,简称SQL)语句的形式发送给服务器,服务器为每个查询生成可执行代码,然后执行查询。编译查询、生成执行计划以及执行查询的任务会增加开销。这些类型的系统存在的另一个问题是:执行不同事务的不同线程可能会尝试同时访问相同的数据。结果,可能会发生大量锁定和锁存冲突,导致性能不佳以及可扩展性较差。

[0006] 其他当前OLTP系统尝试通过采用事务内并行性来提高性能。这些类型的系统使用例如SQL语句在并行执行引擎上的事务中运行每个查询。这些类型的系统存在的问题也包括如上所述的指令数据位置较差和开销增加,以及分析系统性能的困难。

发明内容

[0007] 综上所述,在根据本发明的实施例中,根据动作所使用的数据将事务划分为一组动作,然后将这些动作中的每一个传送给保存该动作待作用的数据集合的节点并由其执行。将线程与数据集合耦合,而不是如上所述将线程与事务耦合。因此,将事务划分为单独路由到存储数据的节点的动作,而不是将数据从分布式节点转到事务。

[0008] 通常,节点可以是设备(例如服务器),或者节点可以在具有其他节点的设备上实例化(例如,可以在单个设备上实现多个节点)。数据在逻辑上和物理上划分为驻留于不同节点(在此称为“执行节点”)上或由其管理的数据集合。在一实施例中,这些集合是不相交的集合。在一实施例中,节点(在此称为“路由节点”)访问或接收对联机事务处理(online transaction processing,简称OLTP)系统中的数据库执行事务的请求。路由节点确定事务待作用于数据库中的哪些数据集合。例如,事务可以作用于第一节点(第一执行节点)上的

第一数据集合，也可以作用于第二节点(第二执行节点)上的第二数据集合。

[0009] 然后根据事务的数据依赖关系将事务分离为多个动作。换句话说，针对事务待作用的每个数据集合建立一个动作。例如，如果事务将作用于两个数据集合，则事务可以分离成第一动作和第二动作，其中第一动作与第一数据集合相关联，第二动作与第二数据集合相关联。在一实施例中，动作是数据操作语言(Data Manipulation Language，简称DML)动作(使用DML语法规定的动作)。动作也可以称为语句、查询、表达式或命令。

[0010] 然后这些动作由路由节点分别传送到存储相应动作待作用的数据的节点(执行节点)。例如，针对第一动作的动作特定消息可以发送给第一执行节点(其存储第一动作待作用的第一数据集合)，针对第二动作的动作特定消息可以发送给第二执行节点(其存储第二动作待作用的第二数据集合)。然后由这些动作路由到的执行节点来执行这些动作。这些动作可以同时并行执行。例如，第一执行节点对第一数据集合执行第一动作，并在相同时间帧内，第二执行节点对第二数据集合执行第二动作。

[0011] 在一实施例中，将每个动作传送给在该动作已经路由到的执行节点上执行的第一线程(在此称为“接收线程”)。例如，将第一动作传送给在第一执行节点上执行的接收线程。然后第一(接收)线程将动作委托给在同一执行节点上执行的动作特定第二线程(在此称为“代理线程”)。第二(代理)线程将动作入队，并可以在动作入队时向第一线程发送消息。第二线程也可以请求并调用该操作待作用的数据集合上的锁。第二线程可以在动作完成时通过向第一线程发送消息来通知第一线程，第一线程转而可以通知路由节点。

[0012] 执行作为事务一部分的动作的每个执行节点在动作完成时通知路由节点。例如，第一执行节点可以在第一动作完成时通知路由节点，第二执行节点可以在第二动作完成时通知路由节点。响应于被通知动作(例如，第一和第二动作)已经完成，路由节点可以调度并发起与事务相关联的下一个(随后的)动作或下一组动作。例如，下一个动作可以是同步刚刚完成的动作的同步动作，或者它可以是作用于OLTP系统中数据库的另一动作(例如，另一DML动作)。一旦与事务相关联的所有动作都已完成，就可以执行提交操作(指示执行节点将完成的动作提交给数据库)。

[0013] 总而言之，根据本发明的实施例使用面向数据的事务处理模型，通过事务内并行性以及事务间并行性，来增加OLTP吞吐量。可以减少冲突的可能性，且可以避免高吞吐量OLTP系统中通常是瓶颈问题的全局锁定。指令数据位置、分析系统性能的能力以及系统可扩展性都得到了改进。基于动作的路由减少了与数据库查询(例如SQL语句)及其执行相关联的开销，从而减少了网络流量并增加了网络带宽。

[0014] 在阅读多个示图描述的下面实施例的具体细节后，本领域普通技术人员将意识到本发明多个实施例的这些以及其他目的和优势。

附图说明

[0015] 附图包含在并且构成本说明书的一部分，其中相同的数字描绘相同的元件，附图说明本发明的实施例，并且与描述内容一起用于解释本发明的原理。

[0016] 图1是示出根据本发明的实施例中的联机事务处理系统中事务的基于动作的路由的示例的框图；

[0017] 图2是示出根据本发明的实施例中的数据分区的示例的框图；

[0018] 图3示出了根据本发明的实施例中的已转换成数据动作和同步动作的事务的示例；

[0019] 图4示出了根据本发明的实施例中的在节点上执行的数据动作的示例；

[0020] 图5是示出根据本发明的实施例中的节点上的数据动作的处理的框图；

[0021] 图6是根据本发明的实施例中的能够用于向节点传送动作的消息的格式的示例；

[0022] 图7是示出根据本发明的实施例中的对事务进行基于动作的路由的由计算机实现的方法的示例的流程图；

[0023] 图8是能够实现根据本发明的实施例的节点的示例的框图。

具体实施方式

[0024] 现将详细地对本发明的各种实施例、附图示出的示例做出参考。虽然会结合这些实施例进行描述，但可以理解的是它们并不用于将本发明限制于这些实施例。相反，本发明旨在覆盖可以包含在由所附权利要求书限定的本发明的精神和范围内的替代物、修改和等同物。另外，在以下本发明的详细描述中，阐述了许多特定细节以便提供对本发明的透彻理解。然而，可以理解的是，实际应用中，可以不包括本发明的这些特定细节。在其它实例中没有详细描述众所周知的方法、流程、部件和电路，以免对本发明的各方面造成不必要地模糊。

[0025] 以下详细描述的一些部分按照过程、逻辑块、处理、以及对计算机存储器内的数据位的操作的其它符号表示来呈现。这些描述和表示是数据处理领域技术人员向所属领域其它技术人员最有效地传达工作实质内容所使用的方法。在本申请案中，将过程、逻辑块、过程或其类似者设想为自相一致的步骤或指令序列，以产生期望的结果。这些步骤利用物理量的物理操作。通常，虽然并不是必须的，但这些量以电信号或磁信号的形式存在，能够被存储、转移、组合和比较，以及在计算机系统中以其它方式被操作。有时候这种方法被证明是便利的，主要出于常见用法的原因，将这些信号称为事务、位、数值、元素、符号、字符、样本、像素或其类似者。

[0026] 然而，应牢记，所有这些和类似术语与适当物理量相关联，且仅为应用于这些量的方便标签。除非在以下讨论中有其他特别声明，应理解，贯穿本发明，使用诸如“访问”、“确定”、“分离”、“路由”、“执行”、“发起”、“指示”、“委托”、“入队”、“发送”、“接收”、“划分”、“锁定”、“通知”、“传送”、“定义”等术语的讨论，可以参考计算系统或类似电子计算设备或处理器(例如，图8的节点810)的动作和过程(例如，图7的流程图700)。计算系统或类似电子计算设备操作并转换在计算系统存储器、寄存器或其它此类信息存储、传输或显示设备内表示为物理(电子)数量的数据。

[0027] 可以在驻留于某一形式的计算机可读存储介质的计算机可执行指令的一般内容背景下讨论本文中所描述的实施例，可执行指令例如是由一个或多个计算机或其它设备执行的程序模块。借助于实例而非限制，计算机可读存储介质可包括非瞬时性计算机存储介质和通信介质。一般而言，程序模块包括执行特定任务或实现特定抽象数据类型的例程、程序、对象、组件以及数据结构等。根据各种实施例中的需要，程序模块的功能可以组合或分布。

[0028] 计算机存储介质包含任何方法或技术中实施用于存储计算机可读指令、数据结

构、程序模块或其它数据等信息的易失性和非易失性、可移动和不可移动介质。计算机存储介质包含但不限于随机存取存储器(random access memory,简称RAM)、只读存储器(read only memory,简称ROM)、电可擦除可编程ROM(electrically erasable programmable ROM,简称EEPROM)、快闪存储器或其它存储器技术、光盘ROM(compact disk ROM,简称CD-ROM)、数字通用光盘(digital versatile disk,简称DVD)或其它光学存储设备、盒式磁带、磁带、磁盘存储设备或其它磁性存储设备、或可用于存储所要信息并可经存取以检索信息的任何其它介质。

[0029] 通信介质可以实施计算机可执行指令、数据结构和程序模块,且包含任何信息递送介质。借助于实例而非限制,通信介质包含有线介质,例如有线网络或直接有线连接;以及无线介质,例如声频、射频(radio frequency,简称RF)、红外线和其它无线介质。以上各项中的任何项的组合也可以包含在计算机可读介质的范围内。

[0030] 图1是示出根据本发明的实施例中的联机事务处理(online transaction processing,简称OLTP)系统100中事务的基于动作的路由的示例的框图。通常,OLTP系统100存储并维护事务数据的数据库。

[0031] 图1的示例系统包括客户端设备110和多个节点,在此称为第一路由节点112、第二路由节点114、第一执行节点116和第二执行节点118。OLTP系统可以包括附加客户端和节点,以及除了所示和所描述的元件之外的元件。这些节点可以统称为集群。节点可以使用基于例如以太网、Infiniband或PCI/e(外围组件互连快递)的连接进行互连。每个节点都可以使用唯一的节点标识(identifier,简称ID)进行标识。

[0032] OLTP系统100可以表征为无共享分布式OLTP系统,其中节点是具有其自己的永久性存储器的多线程系统。在其最基本的配置中,一个节点包括一个处理器、一个存储器和一个与其他节点通信的机制(见图8)。节点可以是诸如计算系统或服务器等的设备。节点也可以是在诸如计算系统或服务器等设备上与其他节点一起实现的元件;即,可以在单个设备上实现多个节点。节点也可以实现为虚拟机。

[0033] 结合图1,路由节点112和114是可以与客户端设备110建立连接并从客户端设备110接受事务(例如,事务130),且可以将事务路由到执行节点116和118的节点。更具体地,在根据本发明的实施例中,路由节点路由从事务中推导出的动作,这将在下面进行更充分地描述。动作也可以称为语句或命令。

[0034] 执行节点116和118是可以执行从路由节点112和114接收的动作的节点。执行节点116和118针对它们存储和维护的数据库的相应部分执行那些动作。即,执行节点116可以存储和维护数据库的第一部分,且可以对第一部分执行动作。执行节点可以存储和维护数据库的第二部分,且可以对第二部分执行动作。路由节点可以是执行节点,执行节点也可以用作路由节点。

[0035] 更具体地,数据库管理系统(database management system,简称DBMS)或分布式DBMS(distributed DBMS,简称DDBMS)可以对OLTP系统100中跨执行节点集群的数据库进行分区。即,由OLTP系统存储和维护的数据库在物理上划分为“数据分区”,每个数据分区存储在相应的执行节点上。执行节点可以存储数据库的不止一个数据分区。在图1的示例中,分区101包括存储在执行节点116上的数据,分区102包括存储在执行节点118上的数据。在一实施例中,数据分区是不相交的数据集合。

[0036] 在一实施例中，数据库布置为表格，每个表格具有一行或多行。在这样的实施例中，可以跨执行节点集群对表格进行物理分区，每个分区可以由分区号或分区ID标识。因此，一个或多个表行的第一分区可以一起存储在一个执行节点上，一个或多个表行(不包括来自第一组的任何行)的第二分区可以存储在另一个执行节点上，等等。存储在一起的一个或多个表行的每个分区可以称为表格分区。在这样的实施例中，表格分区可以在逻辑上划分为一个或多个表格分段。表格分段是表格分区中的表行集合，它们在逻辑上由线程拥有(例如，锁定)。

[0037] 图2是示出根据本发明的实施例中的数据分区的示例的框图。在图2的示例中，数据库200在逻辑上包括多个表行1-8。在该示例中，行1-6构成分区101，行7和行8构成分区102。分区101在物理上存储在第一执行节点116上，分区102在物理上存储在第二执行节点118上。

[0038] 在一实施例中，路由节点(例如，路由节点112)包括将数据(例如，表格分区)映射到其上存储数据集合的执行节点的表格。以下是图2示例中的路由节点112上的映射表的示例，其将分区映射到节点(例如，分区编号/ID映射到节点ID)并且还将行映射到分区(行编号映射到分区编号/ID)。

[0039]

行	分区	节点

[0040]

1-6	101	116

[0041]

7-8	102	118

[0042] 返回结合图1，事务130可以分离为一个或多个动作。一般地，动作是一个可以独立于与事务相关联的其他动作来执行的活动。至少有两种类型的动作与事务相关联：对数据执行的动作以及同步动作。其他类型的动作可能与事务相关联，例如事务管理操作。

[0043] 对数据执行的动作包括在OLTP数据库中添加数据、删除数据或更改(更新)数据的动作。更具体地，这些类型的动作修改(例如，添加、删除、更新)数据库的一部分，例如表格分段。动作也可以称为语句、查询、表达式或命令。为了简化讨论，这些类型的动作在这里将称为数据动作。在一实施例中，数据动作是数据操作语言(Data Manipulation Language，简称DML)动作(使用DML语法规定的动作)。例如，更新数据动作的形式可以是UPDATEtable_name SET column_name＝value[,column_name＝value...][WHERE condition]。在此示例中，表“table_name”中列“column_name”的值将设置为“value”，但仅在那些满足“condition”的行中。图1的动作141、142、143和144是数据动作的示例。

[0044] 同步动作用作事务过程中一次或多次执行的同步点。同步动作实质上将事务划分为多个步骤或时间帧(见图3)。在每个步骤或时间帧内，可以并行执行多个数据动作。实质上，如果两个数据动作之间存在数据依赖关系，那么这些数据动作就会被同步动作分隔开。同步动作将依赖于相同数据的两个数据动作分开，以便事务逐步执行。直到上一步骤的数据动作完成时，事务的下一步骤才开始。

[0045] 与事务关联的最后一个同步动作包括提交操作，特别是两阶段提交操作，以向OLTP数据库提交事务结果(添加、删除、更改)。

[0046] 现在结合图1和图2来描述路由节点112将事务(例如，事务130)转换成数据动作的方式。通常，路由节点112根据动作将作用的数据的位置来定义数据动作。

[0047] 在一实施例中，事务130包括标识事务待修改(添加、删除或更改)的数据集合(例

如，数据分区或分段)的信息。如上所述，路由节点112包括标识数据存储位置的映射表。因此，路由节点112可以为事务待修改的映射表中的每个表项定义一个数据动作。在一实施例中，路由节点112可以基于与数据动作待作用的数据相关联的逻辑分区号/ID来定义数据动作。由于事务中的动作是根据数据所在的位置来定义的，因此可以在不同执行节点上执行该事务。因此，可以在一个执行节点上执行事务130的一些部分，在另一个执行节点上执行其他部分，等等。

[0048] 例如，事务130可以作用于存储在执行节点116上的第一数据集合121和第二数据集合122，还可以作用于存储在执行节点118上的第三数据集合123。在一实施例中，事务130包括标识待作用的每个数据集合的分区ID的信息(例如，第一数据集合121的第一分区ID以及第二数据集合122的第二分区ID等)。因此，路由节点112将事务130分离为：对应于第一数据集合121(例如，对应于第一分区ID)的第一数据动作141；对应于第二数据集合122(例如，对应于第二分区ID)的第二数据动作142；以及对应于第三数据集合123(例如，对应于第三分区ID)的第三数据动作143。在一实施例中，其中数据库200布置为表格，数据集合121、122和123中的每一个对应表格分区(由对应分区ID所标识的表格分区)内的表格分段。

[0049] 通常，可以为事务作用的每个数据集合(例如，表格分段)定义数据动作。由于事务中的动作是根据数据所在的位置来定义的，因此可以在不同执行节点上执行该事务。因此，事务130的一些部分可以在一个执行节点上执行，该事务的其他部分可以在另一个执行节点上执行，等等。

[0050] 然后将每个数据动作传送(路由)到存储该动作待作用的数据的执行节点。因此，在图1的示例中，数据动作141和142路由到第一执行节点116，数据动作143路由到第二执行节点118。

[0051] 然后由这些动作路由到的执行节点来执行这些动作。在图1的示例中，第一执行节点116对第一数据集合121执行数据动作141，对第二数据集合122执行数据动作142；第二执行节点118对第三数据集合123执行数据动作143。

[0052] 在上面的示例中，可以定义作用于执行节点116上两个数据集合121和122的单个数据动作。

[0053] 图3示出了根据本发明的实施例中的已转换成数据动作和同步动作的事务330(类似于图1中的事务130)的示例。在一实施例中，路由节点(例如，图1的路由节点112)从客户端110接收事务330，并将该事务转换成多个数据动作(例如，DML动作)和同步动作。下面进一步讨论路由节点112将事务转换为数据动作和同步动作的方式。

[0054] 在图3的示例中，事务330包括数据动作301、302、303、304和305，还包括同步动作311、312和313。同步动作基本上将事务330划分成多个步骤，每个步骤包括一个或多个数据动作。每个步骤中的数据动作是相互并行执行的。即，它们可能同时也可能不同时执行，但它们是在相同的时间帧内，在相同或不同的执行节点上执行的。

[0055] 因此，在图3的示例中，数据动作301和302相互并行执行。数据动作301和302可以在相同的执行节点上执行，或者它们可以在不同的执行节点上执行，这取决于它们待作用的数据位于何处。一旦数据动作301和302完成，就通知路由节点112，然后执行同步动作311。一旦同步动作311完成，就执行数据动作303和304。一旦数据动作303和304完成，就通知路由节点112，然后执行同步动作312，接下来执行数据动作305。一旦数据动作305完成，

就通知路由节点112，然后执行同步动作313。在图3的示例中，同步动作313是最后一个动作，因此包括提交操作，特别是两阶段提交操作，以将事务330的结果提交给OLTP数据库。

[0056] 图4示出了根据本发明的实施例中的在执行节点上执行的数据动作的示例。在图4的示例中，路由节点112将数据动作141和142传送给执行节点116。

[0057] 在一实施例中，传送到执行节点的每个数据动作由正在该执行节点上执行的第一线程(在此称为“接收线程”)接收，该第一线程转而将该数据动作委托给相应动作特定第二线程(在此称为“代理线程”)。不同的代理线程与每个数据动作相关联。

[0058] 例如，将数据动作141传送到在执行节点116上执行的接收线程401。接收线程401将数据动作141委托给在同一执行节点上执行的动作特定第二线程402(代理线程)。代理线程402将数据动作141入队，且可以在该数据动作入队时向接收线程401发送消息。代理线程402还可以请求并调用数据动作141待作用的数据集合121(例如，由行1和行2组成的表格分段)的锁定。即，数据集合121在逻辑上由代理线程402所拥有。当一行数据锁定时，它不能被另一个线程作用。然后代理线程402根据数据动作141修改数据集合121(例如，添加、删除或更改数据)。当代理线程402处理完数据集合121时，可以解锁该数据。代理线程402还可以在数据动作141完成时通过向接收线程发送消息来通知接收线程401，接收线程401转而可以通知路由节点112。

[0059] 类似地，将数据动作142传送到接收线程401。接收线程401将数据动作142委托给在同一执行节点上执行的另一(第二)代理线程403。第二代理线程403将数据动作142入队，且可以在数据动作142入队时向接收线程401发送消息。第二代理线程403还可以请求并调用数据动作142待作用的数据集合122(例如，由行4组成的表格分段)的锁定。第二代理线程403可以在数据动作142完成时通过向接收线程发送消息来通知接收线程401，接收线程401转而可以通知路由节点112。

[0060] 通过这种方式，执行节点116可以并行执行数据动作141和142。

[0061] 当执行节点116正在执行来自路由节点112的数据动作141和142时，它也可执行来自路由节点112的与除事务130以外的事务相关联的一个或多个其他数据动作。而且，执行节点116可以针对从一个或多个其他路由节点接收的一个或多个事务执行一个或多个数据动作。例如，接收线程401可以从路由节点114接收数据动作144，并将该数据动作委托给动作特定代理线程404，动作特定代理线程404可以在数据动作144入队时向接收线程401发送消息。代理线程404可以调用数据动作144待作用的数据集合124(例如，由行5和6组成的表格分段)的锁定，且可以在数据动作144完成时通知接收线程401，接收线程401转而可以通知路由节点114。

[0062] 以类似的方式，在执行节点118上执行的接收线程可以从路由节点112接收数据动作143，将该数据动作委托给动作特定代理线程，并在数据动作143入队时向该执行节点上执行的接收线程发送消息。该代理线程可以调用数据集合123的锁定，且可以在数据动作143完成时通知执行节点118上的接收线程，接收线程转而可以通知路由节点112。

[0063] 在一实施例中，执行节点包括使用线程ID将数据集合(表格分段或表行)映射到其所有者线程的表格。以下是图4的示例中执行节点116上的逐行映射表的示例。

行	线程ID
1	402
2	402
3	--
4	403
5	404
6	404

[0065] 除了接收线程之外，代理线程的使用通过减少延迟提高了性能。例如，如果接收线程执行委托给代理线程的任务(例如，入队数据动作、请求和调用锁定以及作用于数据的任务)，则在需要执行这些任务的期间接收线程会很忙碌，因此将无法接收另一个数据动作，导致数据动作在执行节点中备份。换句话说，通过使用代理线程，接收线程可以在代理线程执行数据动作时继续从路由节点接收数据动作。

[0066] 图5是示出根据本发明的实施例中的执行节点(例如，执行节点116)上的数据动作(例如，数据动作141)的处理的框图。添加数据动作141到进入队列502(例如，先进先出(first-infirst-out，简称FIFO)缓冲区)。接下来，接收线程401将数据动作141委托给动作特定代理线程402。代理线程402可以访问本地锁定表504(例如上述的映射表)以确定数据动作141待作用的数据集合是否(例如，由正在执行作用于数据动作141待作用的同一数据集合的另一数据动作的另一代理线程)锁定。如果数据动作141待作用的数据被锁定，将该数据动作添加到等待队列506(例如，FIFO缓冲区)，直到该数据解锁。当数据动作141待作用的数据解锁时，可以如上所述由代理线程402执行该数据动作。

[0067] 以类似方式处理其他数据动作。即，将它们添加到进入队列502，依次委托给相应的代理线程并被执行，如果需要的话，添加到等待队列506。

[0068] 图6是根据本发明的实施例中的能够用于路由节点向执行节点传送动作的动作特定消息600的格式的示例。如上所述，事务可以划分成不同的动作，消息600特定于那些动作的其中之一。在图6实施例中，消息600包括由路由节点插入的时间戳和消息序列ID。发送方节点ID字段包括标识路由节点的信息。发送方线程名称/ID字段标识特定操作。接收方节点ID标识该动作路由到的执行节点。在一实施例中，这些字段具有固定长度(它们在消息内处于固定偏移处)，这便于消息的路由。

[0069] 消息600也标识消息类型——待执行的动作的类型。消息的类型包括但不限于：将数据动作发送到执行节点的消息；以及请求执行节点提交的消息。根据消息的类型，消息可以具有不同的长度，消息字段长度标识消息的长度。消息字段包括消息本身。消息结束字段用于指示消息的结束。

[0070] 通常，在根据本发明的实施例中，不是发送整个动作，而是如上所述对动作进行参数化，并将参数化的信息包括在诸如消息600的动作特定消息中。参数化的信息足够允许由接收该消息的执行节点重构该动作。因此，从路由节点发送到执行节点的数据量减少，从而减少了开销和网络流量并增加了带宽。

[0071] 图7是示出根据本发明的实施例中的对事务进行基于动作的路由的由计算机实现

的方法的示例的流程图700。结合图1-4讨论图7。由流程图700中的框表示的全部或部分操作可以实现为驻留于某种形式的非瞬时性计算机可读存储介质上的计算机可执行指令。在流程图700中，一些操作(例如，块702、704、706和712)可以在路由节点(例如，路由节点112和114)上执行，其他操作(例如，块708和710)可以在执行节点(例如，执行节点116和118)上执行。

[0072] 在框702中，在路由节点(例如，路由节点112)处接收或由路由节点访问事务(例如，事务130)。

[0073] 在框704中，路由节点将事务转换为如前所述的数据动作。一般而言，路由节点确定数据库中事务待作用的具体数据集合，然后根据事务的数据依赖关系将事务分离为数据动作。

[0074] 在框706中，将数据动作(例如，数据动作141)传送(路由)到存储数据动作待作用的数据的执行节点(例如，执行节点116)。通常，每个数据动作都路由到存储该数据动作待作用的数据的执行节点。在一实施例中，使用相应的动作特定消息(例如，图6中的消息600)将每个数据动作传送到执行节点。

[0075] 在图7的块708中，在一实施例中，数据动作由第一线程(接收线程)接收，该第一线程在该数据动作已经路由到的执行节点上执行。例如，数据动作141由在第一执行节点116上执行的接收线程401接收。

[0076] 在框710中，在一实施例中，接收线程将数据动作141委托给在同一执行节点上执行的动作特定第二线程(代理线程402)。如上所述，在一实施例中，代理线程402将数据动作入队，在数据动作入队时向接收线程401发送消息，请求并调用数据动作待作用的数据的一个或多个锁定，并在数据动作完成时通过向接收线程发送消息来通知接收线程，接收线程转而通知路由节点112。

[0077] 通常，执行作为事务一部分的数据动作的每个执行节点在数据动作完成时通知路由节点。

[0078] 在框712中，响应于被通知当前一组数据动作已经完成，路由节点可以发起与事务相关联的下一个(随后的)动作或下一组动作。例如，下一个动作可以是同步刚刚完成的动作的同步动作，或者它可以是作用于OLTP系统中数据库的另一动作(例如，另一DML动作)。一旦与事务相关联的所有动作都已完成，就可以执行提交操作(例如，指示执行节点将完成的数据动作提交给数据库的两阶段提交)。

[0079] 因此，根据本发明的实施例使用面向数据的事务处理模型，通过事务内并行性以及事务间并行性，来增加OLTP吞吐量。可以减少冲突的可能性，且可以避免高吞吐量OLTP系统中通常是瓶颈问题的全局锁定。指令数据位置、分析系统性能的能力以及系统可扩展性都得到了改进。基于动作的路由减少了与数据库查询(例如SQL语句)及其执行相关联的开销，从而减少了网络流量并增加了网络带宽。

[0080] 图8是能够实现根据本发明的实施例的节点810的示例的框图。节点810广泛地包括能够执行计算机可读指令的任何单个或多处理器计算设备或系统，诸如结合图7描述的那些。在其最基本的配置中，节点810可以包括至少一个处理电路(例如，处理器814)以及至少一个非易失性存储介质(例如，存储器816)。

[0081] 处理器814通常表示能够处理数据或解释并执行指令的任何类型或形式的处理单

元或电路。在某些实施例中，处理器814可以从软件应用或模块接收指令。这些指令可以使处理器814执行此处描述和/或示出的一个或多个示例实施例的功能。

[0082] 系统存储器816一般表示能够存储数据和/或其他计算机可读指令的任何类型或形式的易失性或非易失性存储设备或介质。系统存储器816的示例包括但不限于RAM、ROM、闪存或任何其他合适的存储设备。尽管不是必需的，但是在某些实施例中，除了非易失性存储单元之外，节点810可以包括易失性存储器单元。

[0083] 除了处理器814和系统存储器816之外，节点810还可以包括一个或多个组件或元件。例如，节点810可以包括存储器控制器、输入/输出(input/output，简称I/O)控制器和通信接口818，其中每个都可以经由通信基础设施互连。

[0084] 通信接口广泛地表示能够利用基于例如以太网、Infiniband和PCI/e的连接来促进节点810和一个或多个附加节点之间的通信的任何类型或形式的通信设备或适配器。

[0085] 节点810可以执行允许其执行操作(例如，图7的操作)的应用840。包含应用840的计算机程序可以加载到节点810中。例如，存储在计算机可读介质上的全部或部分计算机程序可以存储在存储器816中。当由处理器814执行时，计算机程序可以使处理器执行和/或成为用于执行此处描述和/或示出的示例实施例的功能的手段。此外或可选地，此处描述和/或示出的示例实施例可以以固件和/或硬件来实现。

[0086] 许多其他设备或子系统可以连接到节点810。节点810也可以采用任何数量的软件、固件和/或硬件配置。举例来说，可将本文中所公开的实例实施例编码为计算机可读介质上的计算机程序(也称作计算机软件、软件应用、计算机可读指令或计算机控制逻辑)。

[0087] 虽然以上披露使用具体的方框图、流程图以及示例阐明各种实施例，本文中所述和/或图示的每个方框图组件、流程图步骤、操作和/或组件都可以通过各种硬件、软件或固件(或者它们的任意组合)配置单独地和/或共同地实施。另外，对其他组件之中包括的任意组件的披露应该看作为示例，因为可以实施许多其他架构来达到同样的功能。

[0088] 本文中所述和/或图示的进程参数和步骤顺序仅仅是为了举例并且可以按需要更改。例如，虽然本文中所图示和/或描述的步骤可以按照特定顺序来示出或讨论，但这些步骤并非必须按照所图示或所讨论的顺序来执行。本文中所述和/或所图示的各种示例方法还可以省略本文中所述和/或所图示的一个或多个步骤或还可以包括除披露的那些步骤之外的额外步骤。

[0089] 虽然本文已经在全功能性计算系统的背景下对不同的实施例进行了描述和/或图示，这些示例实施例中的一个或多个能够以多种方式作为一个程序产品来分发，而不管用于实际进行该分发的计算机可读介质的具体形式如何。本文中所披露的实施例还可以通过使用执行一些特定任务的软件模块来实施。这些软件模块可以包括脚本、成批文件或其他可执行文件，其中这些可以存储在一种计算机可读介质上或者一种计算系统中。这些软件模块可以配置一个计算系统以用于执行本文中所披露的一个或多个示例实施例。本文中所披露的一个或多个软件模块可以在云计算环境中实施。云计算环境可以通过互联网提供不同的业务和应用。这些基于云的业务(例如，软件即服务、平台即服务、基础设施即服务等等)可以通过网络浏览器或其他远程接口进行访问。本文中所述的各种功能可以通过远程桌面环境或任意其他基于云的计算环境来提供。

[0090] 出于解释的目的，已参考特定实施例描述以上描述内容。然而，以上说明性的讨论

不意图为穷尽的或将本发明限制为所公开的精确形式。根据上述教导，许多修改和变更是可能的。选出和描述的各个实施例的目的是为了更好地解释本发明的原理和其实际应用，因而使本领域技术人员能够更好利用本发明各个实施例和适合预期特定用途的各种变更。

[0091] 因此描述根据本发明的实施例。虽然本发明已经在特定实施例中进行了描述，但是应了解，本发明不应该被解释为被这些实施例限制，而是根据所附权利要求书进行解释。

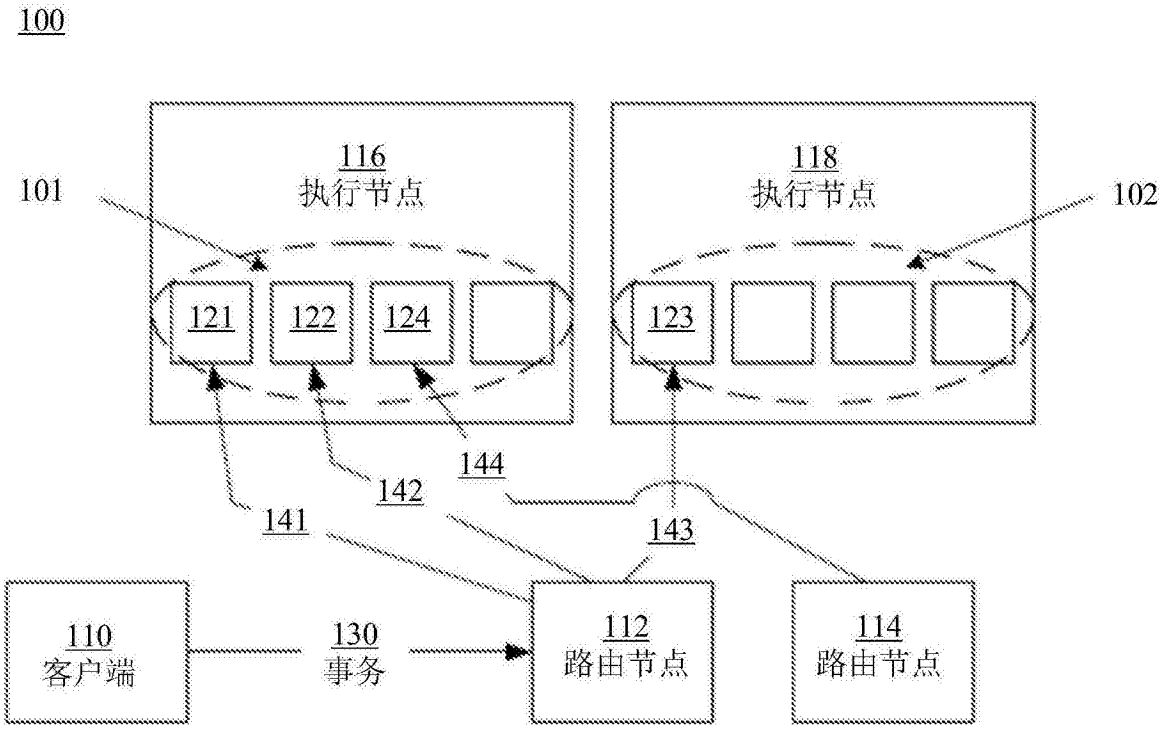

100
101
116
执行节点
118
执行节点
102
121
122
124
123
144
142
141
143
110
客户端
130
事务
112
路由节点
114
路由节点

图1

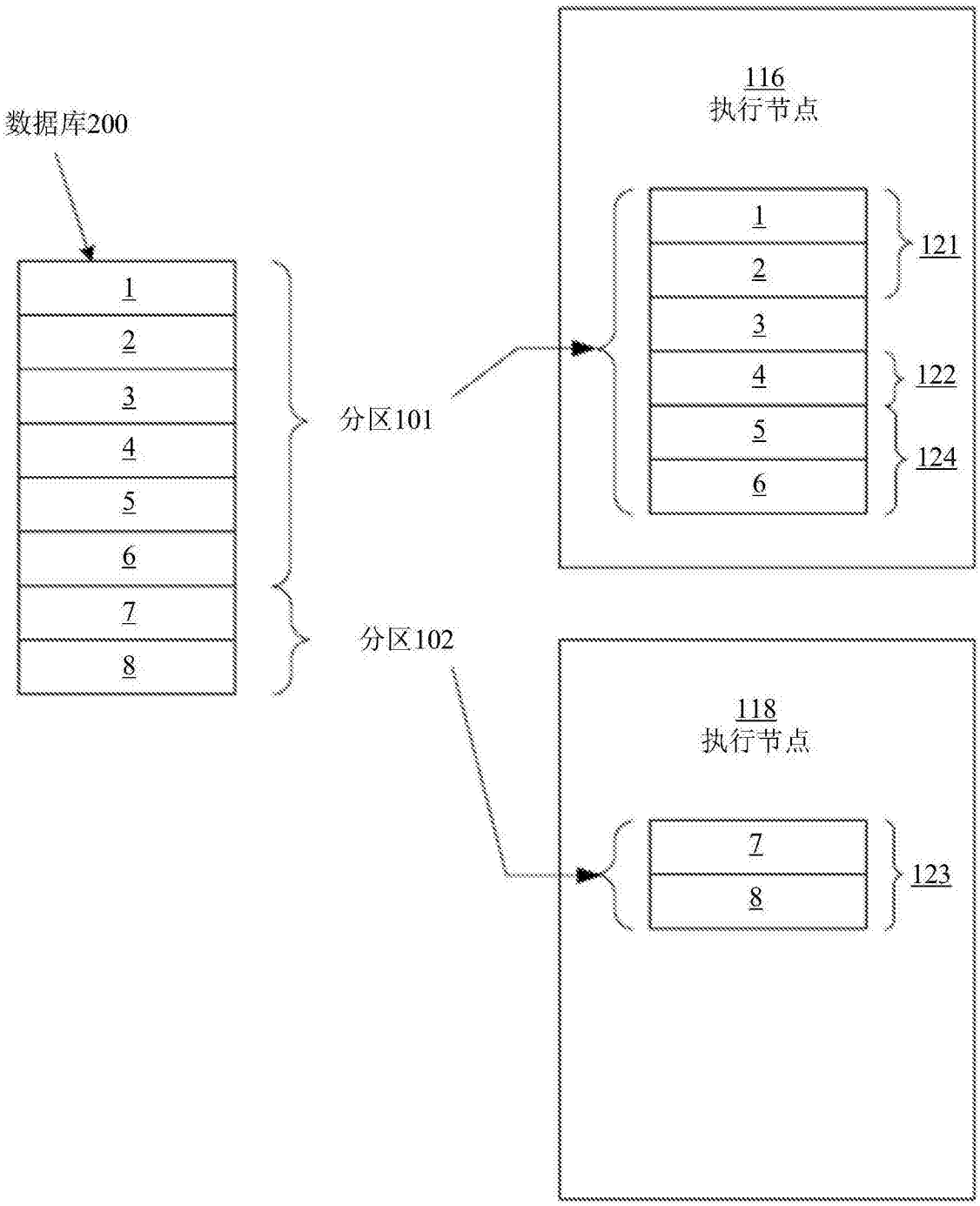

数据库200
1
2
3
4
5
6
7
8
分区101
分区102
116
执行节点
1
2
3
4
5
6
121
122
124
118
执行节点
7
8
123

图2

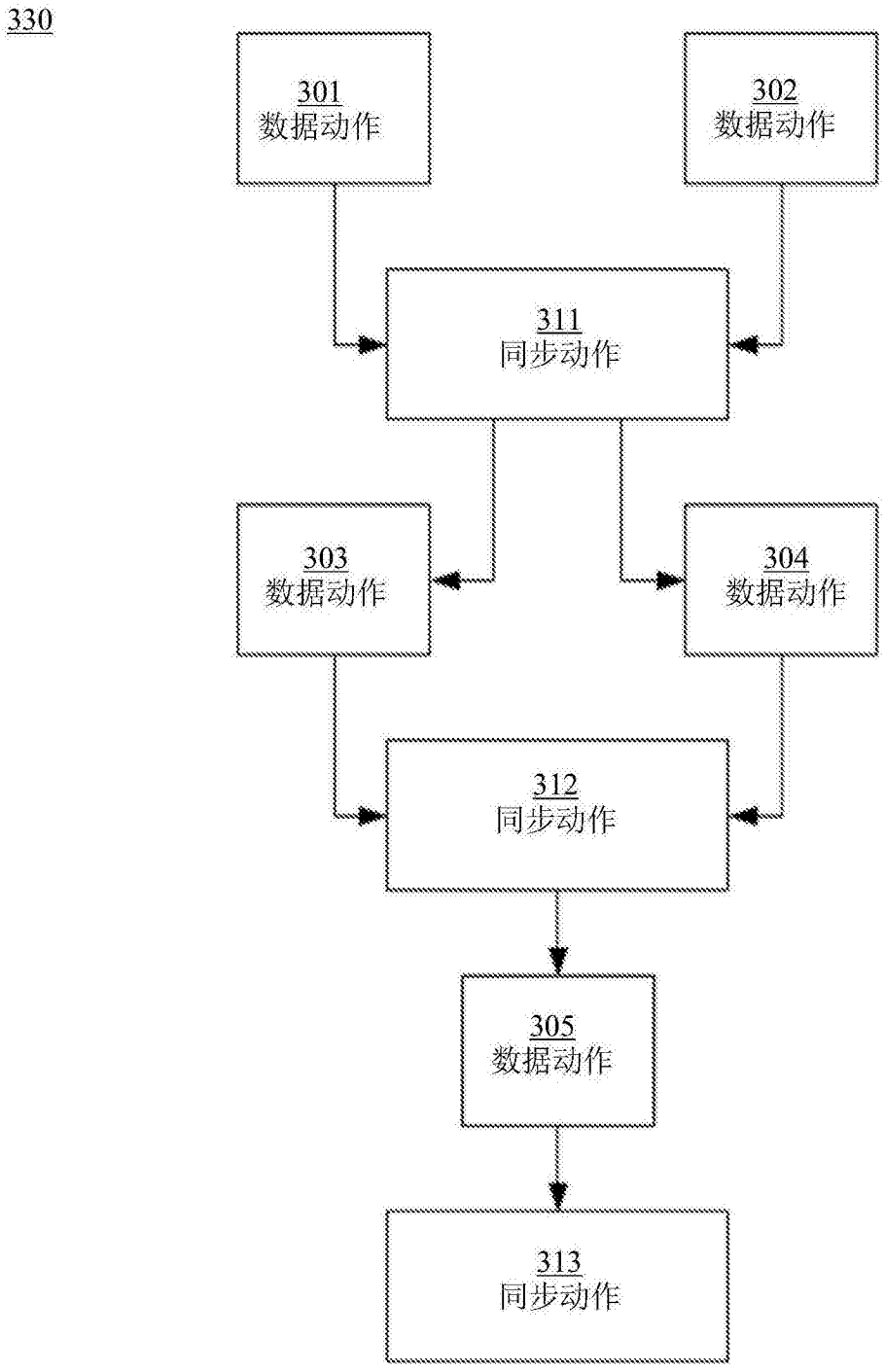

330
301
数据动作
302
数据动作
311
同步动作
303
数据动作
304
数据动作
312
同步动作
305
数据动作
313
同步动作

图3

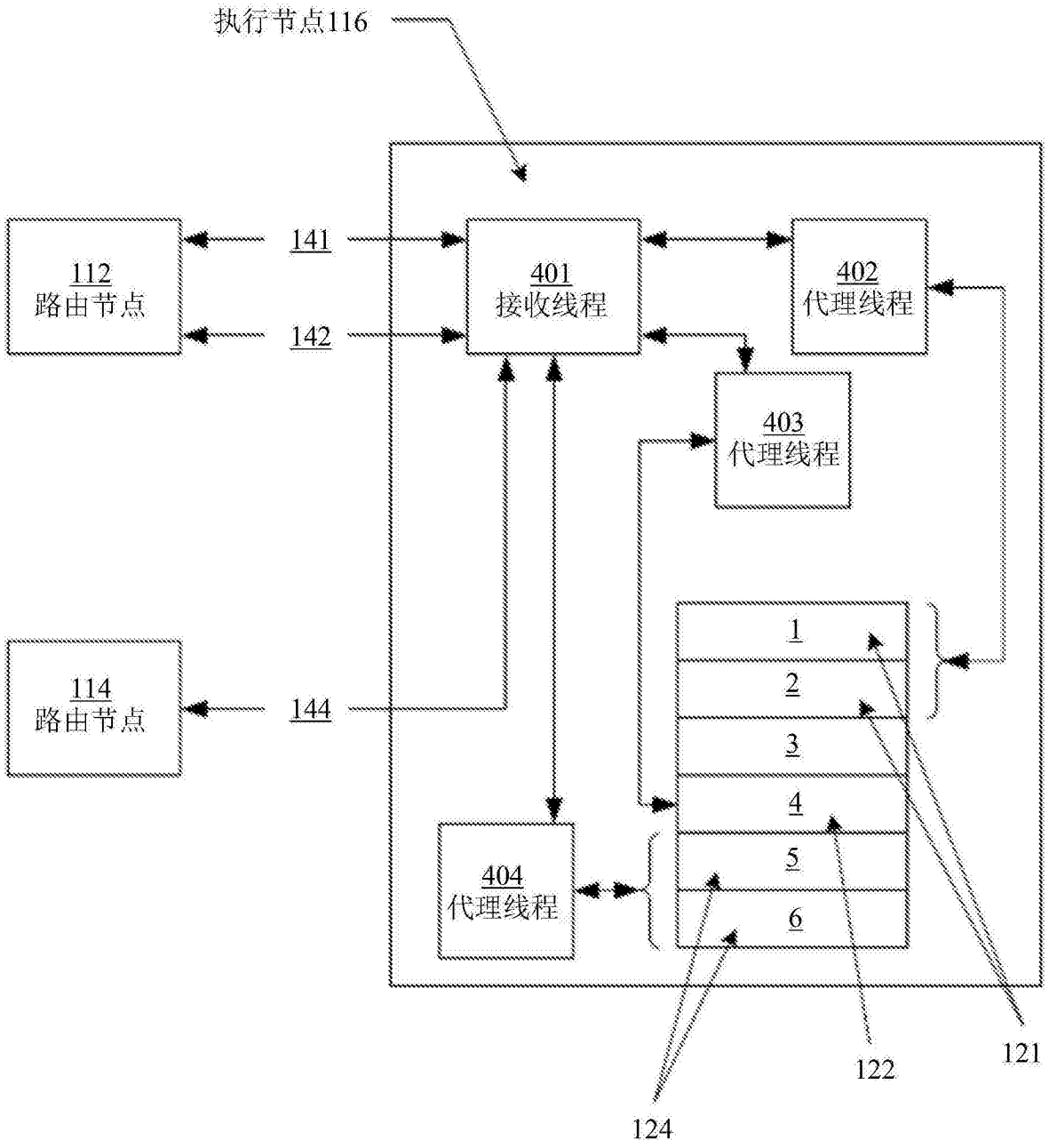

执行节点116
112
路由节点
141
142
401
接收线程
402
代理线程
403
代理线程
114
路由节点
144
1
2
3
4
5
6
404
代理线程
121
122
124

图4

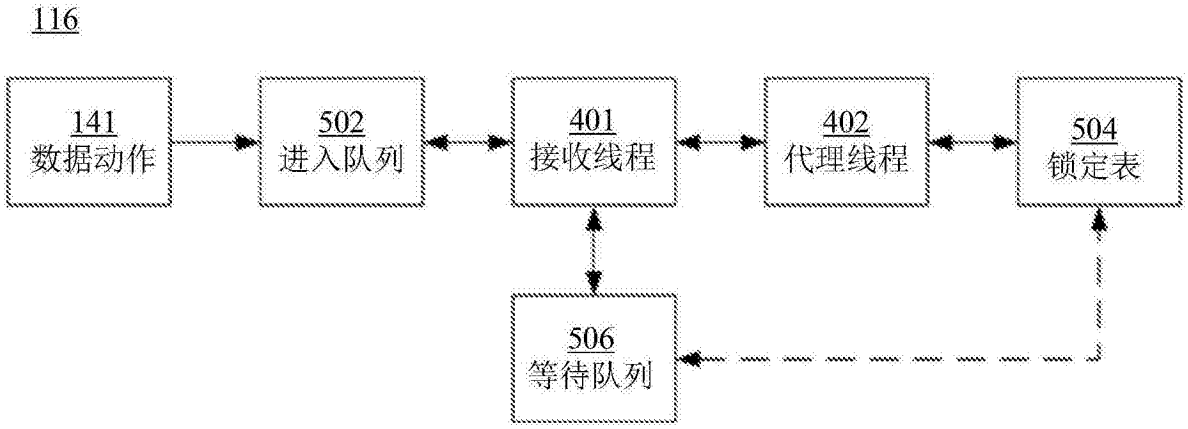

116
141
数据动作
502
进入队列
401
接收线程
402
代理线程
504
锁定表
506
等待队列

图5

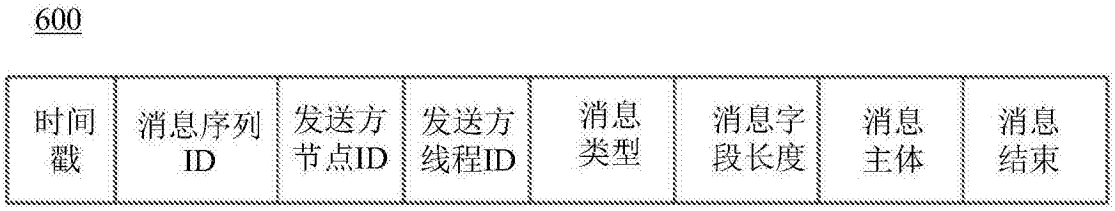

600
时间戳
消息序列ID
发送方节点ID
发送方线程ID
消息类型
消息字段长度
消息主体
消息结束

图6

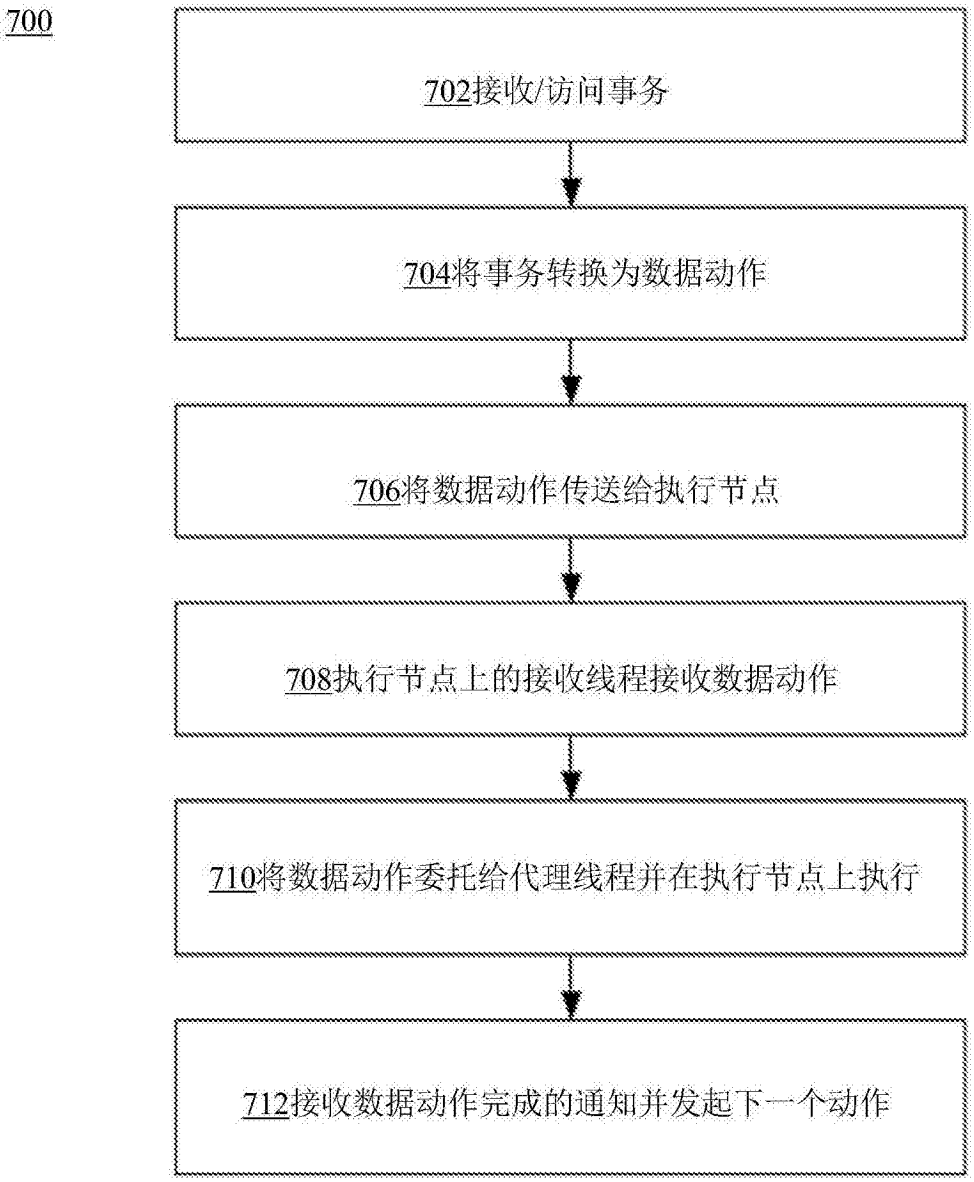

700
702接收/访问事务
704将事务转换为数据动作
706将数据动作传送给执行节点
708执行节点上的接收线程接收数据动作
710将数据动作委托给代理线程并在执行节点上执行
712接收数据动作完成的通知并发起下一个动作

图7

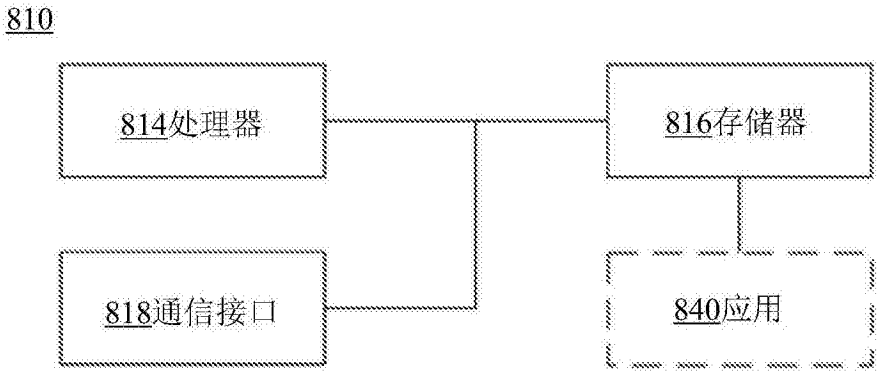

810
814处理器
816存储器
818通信接口
840应用

图8