

区块链的异步落账方法、装置、介质及电子设备

本申请要求于 2020 年 12 月 28 日提交中国专利局、申请号为 202011577543.1、申请名称为“区块链的异步落账方法、装置、介质及电子设备”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

5 技术领域

本申请涉及计算机及通信技术领域，具体涉及区块链的异步落账。

背景技术

10 区块链网络在对区块达成共识之后，区块链节点需要将区块信息写入账本中，通常节点记账方法是将区块信息串行写入磁盘或者分布式存储，例如先将区块头写入磁盘或者分布式存储，等该步骤完成后继续将交易列表以及交易执行结果集写入磁盘或者分布式存储，等交易列表和交易执行结果集写完后，再将写集合写入磁盘或者分布式存储，当写集合写完后节点记账完成开始下一轮共识出块。

发明内容

15 本申请的实施例提供了一种区块链的异步落账方法、装置、介质及电子设备，进而至少在一定程度上可以降低区块的落账延迟，提升区块链网络的处理性能。

本申请的其他特性和优点将通过下面的详细描述变得显然，或部分地通过本申请的实践而习得。

20 根据本申请实施例的一个方面，提供了一种区块链的异步落账方法，包括：在区块达成共识之后，将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列中；在将所述各项数据分别写入对应的消息队列中之后，开启下一轮的区块共识过程；分别从所述各项数据对应的消息队列中读取数据进行区块链账本的持久化处理。

25 根据本申请实施例的一个方面，提供了一种区块链的异步落账装置，包括：第一处理单元，配置为在区块达成共识之后，将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列中；第二处理单元，配置为在将所述各项数据分别写入对应的消息队列中之后，开启下一轮的区块共识过程；持久化单元，配置为分别从所述各项数据对应的消息队列中读取数据进

行区块链账本的持久化处理。

在本申请的一些实施例中，基于前述方案，所述第一处理单元还配置为：将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列及缓存中。

- 5 在本申请的一些实施例中，基于前述方案，所述第一处理单元还配置为：在所述区块达成共识之后，将所述区块及对应的区块执行结果写入日志中；针对所述日志中包含的目标区块，在对所述目标区块对应的各项数据进行持久化处理之后，从所述日志中删除所述目标区块及所述目标区块的区块执行结果。

- 10 在本申请的一些实施例中，基于前述方案，所述第一处理单元还配置为：在区块链节点重启后，获取所述日志中所包含的区块及区块执行结果；根据从所述日志中获取到的区块及区块执行结果进行持久化处理。

在本申请的一些实施例中，基于前述方案，所述第一处理单元配置为：在将所述区块及对应的区块执行结果成功写入所述日志之后，将所述区块所对应区块信息中所包含的各项数据分别写入所述各项数据对应的消息队列中。

- 15 在本申请的一些实施例中，基于前述方案，所述第一处理单元还配置为：若将所述区块及对应的区块执行结果写入所述日志失败，则进行重试处理，当重试次数达到设定次数后仍未成功写入所述日志，停止对所述区块进行落账处理并发出告警通知消息。

- 20 在本申请的一些实施例中，基于前述方案，所述第二处理单元还配置为：在对目标区块对应的各项数据进行持久化处理之后，从所述缓存中删除所述目标区块对应的各项数据。

- 25 在本申请的一些实施例中，基于前述方案，所述第一处理单元配置为：将所述区块信息中所包含的区块头写入区块头对应的队列中及缓存中，将所述区块信息中所包含的交易数据写入交易数据对应的队列中及缓存中，将所述区块信息中所包含的写集合写入写集合对应的队列中及缓存中，其中，所述交易数据包括交易信息列表及交易执行结果。

在本申请的一些实施例中，基于前述方案，所述区块信息中包含的各项数据包括：区块头、交易数据及写集合；所述第一处理单元配置为：以区块高度作为关键字、区块头数据作为值，将所述区块信息中包含的区块头写入所述区

块头对应的缓存中；以交易哈希作为关键字、交易数据作为值，将所述区块信息中包含的交易数据写入所述交易数据对应的缓存中；以表名和索引作为关键字、写数据作为值，将所述区块信息中包含的写集合写入所述写集合对应的缓存中。

- 5 在本申请的一些实施例中，基于前述方案，所述持久化单元配置为：对所述各项数据对应的消息队列进行监听，目标消息队列为所述各项数据对应的消息队列中的一个消息队列；若监听到所述目标消息队列中包含有数据，则从所述目标消息队列中依次读取数据进行持久化处理。

- 10 在本申请的一些实施例中，基于前述方案，所述持久化单元配置为：调用存储层接口对从所述目标消息队列中依次读取出的数据进行持久化处理；若持久化处理失败，则重新调用所述存储层接口对从所述目标消息队列中读取出的数据进行持久化处理，直到持久化处理成功。

- 15 在本申请的一些实施例中，基于前述方案，所述持久化单元配置为：启动针对各个所述消息队列的监听任务；通过针对各个所述消息队列的监听任务，对所述各项数据对应的消息队列进行监听。

在本申请的一些实施例中，基于前述方案，所述第二处理单元还配置为：在对所述区块信息完成持久化处理之前，通过所述缓存中存储的数据对业务请求进行响应。

- 20 根据本申请实施例的一个方面，提供了一种计算机可读介质，其上存储有计算机程序，所述计算机程序被处理器执行时实现如上述实施例中所述的区块链的异步落账方法。

- 25 根据本申请实施例的一个方面，提供了一种电子设备，包括：一个或多个处理器；存储装置，用于存储一个或多个程序，当所述一个或多个程序被所述一个或多个处理器执行时，使得所述一个或多个处理器实现如上述实施例中所述的区块链的异步落账方法。

根据本申请实施例的一个方面，提供了一种计算机程序产品或计算机程序，该计算机程序产品或计算机程序包括计算机指令，该计算机指令存储在计算机可读存储介质中。计算机设备的处理器从计算机可读存储介质读取该计算机指令，处理器执行该计算机指令，使得该计算机设备执行上述各种可选实施例中

提供的区块链的异步落账方法。

在本申请的一些实施例所提供的技术方案中，在区块达成共识之后，将达成共识的区块所对应的区块信息中包含的各项数据分别写入对应的消息队列中，并在将各项数据分别写入对应的消息队列中之后，开启下一轮的区块共识过程，然后分别从各项数据对应的消息队列中读取数据进行区块链账本的持久化处理。可见，本申请实施例的技术方案一方面通过并行的对区块信息中包含的各项数据进行处理，降低了区块的落账延迟；另一方面在将数据写入消息队列中之后就可以开启下一轮的区块共识过程，无需等待区块中的各项数据的持久化过程，提高了区块链网络的吞吐量，进而有效提升了区块链网络的处理性能。

应当理解的是，以上的一般描述和后文的细节描述仅是示例性和解释性的，并不能限制本申请。

附图说明

图1示出了区块链网络的结构示意图；
图2示出了区块链中各区块的连接关系示意图；
图3示出了生成一个区块的过程示意图；
图4示出了根据本申请的一个实施例的区块链的异步落账方法的流程图；
图5示出了根据本申请的一个实施例的区块链的异步落账装置的框图；
图6示出了适于用来实现本申请实施例的电子设备的计算机系统的结构示意图。

具体实施方式

现在将参考附图更全面地描述示例实施方式。然而，示例实施方式能够以多种形式实施，且不应被理解为限于在此阐述的范例；相反，提供这些实施方式使得本申请将更加全面和完整，并将示例实施方式的构思全面地传达给本领域的技术人员。

此外，所描述的特征、结构或特性可以以任何合适的方式结合在一个或更多实施例中。在下面的描述中，提供许多具体细节从而给出对本申请的实施例的充分理解。然而，本领域技术人员将意识到，可以实践本申请的技术方案而没有特定细节中的一个或更多，或者可以采用其它的方法、组元、装置、步骤

等。在其它情况下，不详细示出或描述公知方法、装置、实现或者操作以避免模糊本申请的各方面。

附图中所示的方框图仅仅是功能实体，不一定必须与物理上独立的实体相对应。即，可以采用软件形式来实现这些功能实体，或在一个或多个硬件模块或集成电路中实现这些功能实体，或在不同网络和/或处理器装置和/或微控制器装置中实现这些功能实体。

附图中所示的流程图仅是示例性说明，不是必须包括所有的内容和操作/步骤，也不是必须按所描述的顺序执行。例如，有的操作/步骤还可以分解，而有的操作/步骤可以合并或部分合并，因此实际执行的顺序有可能根据实际情况改变。

需要说明的是：在本文中提及的“多个”是指两个或两个以上。“和/或”描述关联对象的关联关系，表示可以存在三种关系，例如，A和/或B可以表示：单独存在A，同时存在A和B，单独存在B这三种情况。字符“/”一般表示前后关联对象是一种“或”的关系。

区块链（Blockchain）是分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式。区块链本质上是一个去中心化的数据库，是一串使用密码学方法相关联产生的数据块（即区块），每一个数据块中包含了一批网络交易的信息，用于验证其信息的有效性（防伪）和生成下一个区块。区块链可以包括区块链底层平台、平台产品服务层以及应用服务层。

区块链底层平台可以包括用户管理、基础服务、智能合约以及运营监控等处理模块。其中，用户管理模块负责所有区块链参与者的身份信息的管理，包括维护公私钥生成（账户管理）、密钥管理以及用户真实身份和区块链地址对应关系维护（权限管理）等，并且在授权的情况下，监管和审计某些真实身份的交易情况，提供风险控制的规则配置（风控审计）；基础服务模块部署在所有区块链节点设备上，用来验证业务请求的有效性，并对有效请求完成共识后记录到存储上，对于一个新的业务请求，基础服务先对接口适配解析和鉴权处理（接口适配），然后通过共识算法将业务信息加密（共识管理），在加密之后完整一致的传输至共享账本上（网络通信），并进行记录存储；智能合约模块负责合约的注册发行以及合约触发和合约执行，开发人员可以通过某种编程语言

-6-

定义合约逻辑，发布到区块链上（合约注册），根据合约条款的逻辑，调用密钥或者其它的事件触发执行，完成合约逻辑，同时还提供对合约升级注销的功能；运营监控模块主要负责产品发布过程中的部署、配置的修改、合约设置、云适配以及产品运行中的实时状态的可视化输出，例如：告警、监控网络情况、

5 监控节点设备健康状态等。

平台产品服务层提供典型应用的基本能力和实现框架，开发人员可以基于这些基本能力，叠加业务的特性，完成业务逻辑的区块链实现。应用服务层提供基于区块链方案的应用服务给业务参与方进行使用。

参见图1所示的区块链网络，该区块链网络中可以包括多个节点101，多个节点101可以是形成区块链网络的各个客户端。每个节点101在进行正常工作可以接收到输入信息，并基于接收到的输入信息维护该区块链网络内的共享数据。为了保证区块链网络内的信息互通，区块链网络中的每个节点之间可以存在信息连接，节点之间可以通过上述信息连接进行信息传输。例如，当区块链网络中的任意节点接收到输入信息时，区块链网络中的其它节点便根据共识算法获取该输入信息，将该输入信息作为共享数据进行存储，使得区块链网络中全部节点上存储的数据均一致。

对于区块链网络中的每个节点，均具有与其对应的节点标识，而且区块链网络中的每个节点均可以存储有其它节点的节点标识，以便后续根据其它节点的节点标识，将生成的区块广播至区块链网络中的其它节点。每个节点中可维护一个节点标识列表，将节点名称和节点标识对应存储至该节点标识列表中。其中，节点标识可为IP（Internet Protocol，网络之间互联的协议）地址以及其他任一种能够用于标识该节点的信息。

区块链网络中的每个节点均存储一条相同的区块链。区块链由多个区块组成，参见图2所示，区块链由多个区块组成，创始块中包括区块头和区块主体，区块头中存储有输入信息特征值、版本号、时间戳和难度值等，区块主体中存储有输入信息；创始块的下一区块以创始块为父区块，下一区块中同样包括区块头和区块主体，区块头中存储有当前区块的输入信息特征值、父区块的区块头特征值、版本号、时间戳和难度值等，并以此类推，使得区块链中每个区块中存储的区块数据均与父区块中存储的区块数据存在关联，保证了区块中输入

信息的安全性。

在生成区块链中的各个区块时，参见图3所示，区块链所在的节点在接收到输入信息时，对输入信息进行校验，完成校验后，将输入信息存储至内存池中，并更新其用于记录输入信息的哈希树；之后，将更新时间戳更新为接收到输入信息的时间，并尝试不同的随机数，多次进行特征值计算，使得计算得到的特征值可以满足下述公式：

$$SHA256(SHA256(version + prev_hash + merkle_root + ntime + nbits + x)) < TARGET$$

其中，SHA256为计算特征值所用的特征值算法；version（版本号）为区块链中相关区块协议的版本信息；prev_hash为当前区块的父区块的区块头特征值；merkle_root为输入信息的特征值；ntime为更新时间戳的更新时间；nbits为当前难度，在一段时间内为定值，并在超出固定时间段后再次进行确定；x为随机数；TARGET为特征值阈值，该特征值阈值可以根据nbits确定得到。

这样，当计算得到满足上述公式的随机数时，便可将信息对应存储，生成区块头和区块主体，得到当前区块。随后，区块链所在节点根据数据共享系统中其它节点的节点标识，将新生成的区块分别发送给其所在的数据共享系统中的其它节点，由其它节点对新生成的区块进行校验，并在完成校验后将新生成的区块添加至其存储的区块链中。

区块链网络中各个节点可以是服务器，也可以是终端设备。服务器可以是独立的物理服务器，也可以是多个物理服务器构成的服务器集群或者分布式系统，还可以是提供云服务、云数据库、云计算（Cloud Computing）、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、CDN（Content Delivery Network，内容分发网络）、以及大数据和人工智能平台等基础云计算服务的云服务器。终端设备可以是智能手机、平板电脑、笔记本电脑、台式计算机、智能音箱、智能手表等，但并不局限于此。各个节点之间可以通过有线或无线通信方式进行直接或间接地连接，本申请在此不做限制。

前述的云计算是指IT基础设施的交付和使用模式，具体指通过网络以按需、易扩展的方式获得所需资源；广义云计算指服务的交付和使用模式，指通过网络以按需、易扩展的方式获得所需服务。这种服务可以是IT和软件、互联网相关，也可是其他服务。云计算是网格计算（Grid Computing）、分布式计算

(Distributed Computing)、并行计算 (Parallel Computing)、效用计算 (Utility Computing)、网络存储 (Network Storage Technologies)、虚拟化 (Virtualization)、负载均衡 (Load Balance) 等传统计算机和网络技术发展融合的产物。随着互联网、实时数据流、连接设备多样化的发展, 以及搜索服务、社会网络、移动商务和开放协作等需求的推动, 云计算迅速发展起来。不同于以往的并行分布式计算, 云计算的产生从理念上将推动整个互联网模式、企业管理模式发生革命性的变革。

其中, 区块链网络在对区块达成共识之后, 区块链节点需要将区块信息写入账本中, 该区块信息包括区块以及区块执行结果, 区块包括区块头、交易列表; 区块执行结果包括执行交易结果集以及写集。

通常节点记账方法是将区块信息串行写入磁盘或者分布式存储, 例如先将区块头写入磁盘或者分布式存储, 等该步骤完成后继续将交易列表以及交易执行结果集写入磁盘或者分布式存储, 等交易列表和交易执行结果集写完后, 再将写集合写入磁盘或者分布式存储, 当写集合写完后节点记账完成开始下一轮共识出块。但是, 由于这种区块落账的方案整体采用串行方式, 性能严重依赖 I/O 资源, 导致每一轮区块落账延迟较大。

基于此, 本申请的实施例提出了一种新的区块链的异步落账方案, 以下对本申请实施例的技术方案的实现细节进行详细阐述:

图4示出了根据本申请的一个实施例的区块链的异步落账方法的流程图, 该区块链的异步落账方法可以由区块链网络中的任意一个共识节点来执行。参照图4所示, 该区块链的异步落账方法至少包括S410至S430, 详细介绍如下:

在S410中, 在区块达成共识之后, 将区块所对应区块信息中包含的各项数据分别写入各项数据对应的消息队列中。

在本申请的一个实施例中, 可以为区块信息中包含的各项数据分别设置一个消息队列, 其中的消息队列用于存放需要进行持久化处理的数据。

同时, 在一种可能的实现方式中, 还可以为各项数据分别设置一个缓存, 该缓存的作用是在对区块信息完成持久化处理之前, 可以通过缓存中存储的数据对业务请求进行响应, 比如通过缓存中存储的数据对查询请求进行响应等。

在本申请的一个实施例中, 区块信息中包含的各项数据可以包括: 区块头、

交易数据及写集合。其中，区块头中存储有区块的头信息，包含上一个区块的哈希值（PreHash）、本区块体的哈希值（Hash），以及时间戳（TimeStamp）等等。交易数据包括交易信息列表及交易执行结果消息。写集合中包含了背书节点在预执行交易的过程中对变量预写入的值，即该对变量的写入只是记录在写集合中，并未更改世界状态，写集合中也以键值对的形式进行记录。

基于前述实施例中区块信息所包含的各项数据，可以将区块头写入区块头对应的队列中及缓存中，将交易数据写入交易数据对应的队列中及缓存中，将写集合写入写集合对应的队列中及缓存中。

在本申请的一个实施例中，可以以区块高度作为关键字、区块头数据作为值，将区块信息中包含的区块头写入区块头对应的缓存中。

在本申请的一个实施例中，以交易哈希作为关键字、交易数据作为值，将区块信息中包含的交易数据写入交易数据对应的缓存中。

在本申请的一个实施例中，以表名和索引作为关键字、写数据作为值，将区块信息中包含的写集合写入写集合对应的缓存中。

继续参照图4所示，在S420中，在将各项数据分别写入对应的消息队列中之后，开启下一轮的区块共识过程。

在本申请的一个实施例中，当把区块信息中包含的各项数据分别写入对应的消息队列中之后，或者当把区块信息中包含的各项数据分别写入对应的消息队列及缓存中之后，区块链网络中的各个共识节点可以开启下一轮的区块共识过程，无需等待区块信息进行持久化处理之后再开启下一轮的共识过程，有效提高了区块链网络的共识性能。

继续参照图4所示，在S430中，分别从各项数据对应的消息队列中读取数据进行区块链账本的持久化处理。

在本申请的一个实施例中，可以对各项数据对应的消息队列进行监听，目标消息队列为所述各项数据对应的消息队列中的一个消息队列。如果监听到目标消息队列中包含有数据，则从目标消息队列中依次读取数据进行持久化处理。其中，目标消息队列是区块信息中的各项数据所对应的消息队列中包含有数据的消息队列。可选地，可以启动针对各个消息队列的监听任务，然后通过针对各个消息队列的监听任务，对各项数据对应的消息队列进行监听。

在本申请的一个实施例中,从目标消息队列中依次读取数据进行持久化处理的过程可以是调用存储层接口对从目标消息队列中依次读取出的数据进行持久化处理。如果持久化处理失败,则重新调用存储层接口对从目标消息队列中读取出的数据进行持久化处理,直到持久化处理成功。

5 在本申请的一个实施例中,在对目标区块对应的各项数据进行持久化处理之后,可以从缓存中删除目标区块对应的各项数据。该实施例的技术方案可以保证在区块对应的各项数据持久化处理之后,及时清理缓存,以释放缓存的存储空间。

10 基于图4所示实施例的技术方案,在本申请的一个实施例中,在区块达成共识之后,可以将达成共识的区块及对应的区块执行结果写入日志中,针对所述日志中包含的目标区块,可以在对目标区块对应的各项数据进行持久化处理之后,从日志中删除目标区块及目标区块的区块执行结果。该实施例的技术方案使得可以通过日志暂存达成共识的区块及对应的区块执行结果,进而使得区块链节点能够基于该日志确定是否有未进行持久化处理的区块信息,进而可以
15 保证各区块链节点维护的区块高度一致。

在本申请的一个实施例中,基于前述实施例中的日志,使得区块链节点在重启后,可以获取日志中所包含的区块及区块执行结果,然后根据从日志中获取到的区块及区块执行结果进行持久化处理。该实施例的技术方案使得区块链节点在重启之后,能够基于对该日志的读取来确保所存储的区块信息的完整性。

20 作为本申请的一个实施例,当区块链网络中的区块达成共识之后,可以先将达成共识的区块及对应的区块执行结果写入日志,并且在写入日志成功之后,再将达成共识的区块所对应的区块信息中所包含的各项数据分别写入各项数据对应的消息队列中,或也写入缓存中。即在该实施例中,写入日志的优先级较高,只有在写入日志成功之后,再将区块信息中所包含的各项数据分别写入
25 各项数据对应的消息队列中,或也写入缓存中。同时,在本申请的一个实施例中,若将达成共识的区块及对应的区块执行结果写入日志失败,则进行重试处理,当重试次数达到设定次数后仍未成功写入所述日志,停止进行落账处理并发出告警通知消息。

在本申请的一个实施例中,当区块链网络中的区块达成共识之后,将达成

共识的区块及对应的区块执行结果写入日志的过程, 以及将达成共识的区块所对应的区块信息中所包含的各项数据分别写入各项数据对应的消息队列及缓存中的过程也可以是同时进行的; 或者也可以先写入消息队列及缓存中, 再写入日志中。并且可以在写入失败之后, 都进行重试, 当重试次数达到阈值之后,

5 进行报警通知处理。

以下以一个具体的实施例, 对本申请的实现细节进行详细阐述:

在本申请的一个实施例中, 可以设置一个日志管理组件, 该日志管理组件主要用于当节点在重启后进行重新落账时, 防止账本数据与其它节点数据不一致。具体而言, 该日志管理组件会保存若干个最新达成共识的区块以及区块执行结果, 节点重启后账本模块会首先读取该日志管理组件中的区块以及区块执行结果, 然后将读取到的数据重新写入磁盘或者分布式存储, 以保证本节点数据的完备性以及与其它节点在相同区块高度上数据状态一致性。同时, 日志管理组件中存储的区块及区块执行结果会进行删除, 具体是节点在将某个区块的相关数据进行持久化处理之后, 可以从该日志管理组件中删除已经进行了持久化处理的区块及区块执行结果。

在本申请的一个实施例中, 可以设置三个不同的队列组件, 一个队列为block queue用来存放区块头数据; 一个队列为transaction queue用来存放交易数据 (即交易信息列表以及交易执行结果); 一个队列为writeset queue用来存放写集合。同时引入了三个任务, 一个任务为persist block job用来实时监听block queue是否有区块头数据写入; 一个任务为persist transaction job用来实时监听transaction queue是否有交易数据写入; 一个任务为persist writeset job用来实时监听writeset queue是否有写集合写入。

在本申请的一个实施例中, 还可以引入缓存管理组件, 该缓存管理组件分别缓存和清理block cache中的区块头数据、transaction cache中的交易数据以及writeset cache中的写集合数据。其中, block cache用区块高度为关键字、区块头数据为值进行缓存; transaction cache用交易哈希为关键字、交易数据为值进行缓存; writeset cache用表名加索引为关键字、写数据为值进行缓存。

在本申请的一个实施例中, 账本模块被调用写入新的区块数据时, 首先将区块以及区块执行结果写入日志管理组件中, 如果写入日志管理组件失败, 则

账本模块基于失败原因决定是否重试,当重试达到一定次数后终止该区块的落账处理,并通知相关维护人员进行检查修复。

在本申请的一个实施例中,账本模块在将区块数据写入日志管理组件中之后,将采用并发执行逻辑分别将区块头、交易数据(即交易信息列表以及交易执行结果)、写集合同同时写入操作。具体的写入操作如下,写入区块头数据会按照block queue的规范(如格式要求等)生成一个区块头消息写入block queue,然后按照缓存管理组件block cache的要求写入缓存;写入交易数据同理按照transaction queue的规范生成一个交易列表消息写入transaction queue,然后按照transaction cache的要求写入缓存;写入写集合同理按照writeset queue的规范写入队列,然后将写集合写入writeset cache。当上述三类数据分别写入队列和缓存中之后,可以认为记账结束,此时可以开启下一轮的区块共识过程。由于上述三类数据采用并发写入,记账延迟以其中某一类数据写入延迟为准,大大降低了记账延迟。

为了持久化账本数据,上述的三个实时任务分别对三个队列进行监听, persist block job监听block queue是否有区块头消息需要写入存储, persist transaction job监听transaction queue是否有交易数据需要写入存储, persist writeset job监听writeset queue是否有写集合数据需要写入存储。

如果persist block job任务监听到block queue有区块头数据需要写入存储,则从block queue依次读取区块头数据,由于区块头数据是按照区块高度顺序放入block queue的,所以该任务读取出来的区块头数据也是按照区块高区顺序,不需要额外的排序处理,当任务读取到区块头数据后,调用存储层接口持久化区块数据。类似地,如果persist transaction job任务监听到transaction queue有交易数据需要写入存储,则从transaction queue依次读取交易数据,当任务读取到交易数据后,调用存储层接口持久化区块数据。如果persist writeset job任务监听到writeset queue有写集合数据需要写入存储,则从writeset queue依次读取写集合数据,当任务读取到写集合数据后,调用存储层接口持久化区块数据。

为了持久化数据,上述的三个任务需要保证调用存储层持久化过程正确返回,如果尝试持久化失败后重新调用存储层接口重试,重试的过程能保证数据的幂等性,不管尝试多少次,存储的数据和第一次持久化后的数据一致。

在本申请的一个实施例中，如上所述，当对某个区块的相关数据进行持久化处理之后，可以从日志管理组件中删除已经进行了持久化处理的区块及区块执行结果，进而及时对日志管理组件中存储的数据进行清理。同时，当对某个区块的相关数据进行持久化处理之后，可以从缓存中删除已经进行了持久化处理的区块所对应的各项数据，以及及时清理block cache中的区块头数据、transaction cache中的交易数据以及writeset cache中的写集合数据。

本申请上述实施例的技术方案通过并行的对区块信息中包含的各项数据进行处理，降低了区块的落账延迟；另一方面在将数据写入消息队列及缓存中之后就可以开启下一轮的区块共识过程，无需等待区块中的各项数据的持久化过程，实现了异步落账的过程，提高了区块链网络的吞吐量，进而有效提升了区块链网络的处理性能。

以下介绍本申请的装置实施例，可以用于执行本申请上述实施例中的区块链的异步落账方法。对于本申请装置实施例中未披露的细节，请参照本申请上述的区块链的异步落账方法的实施例。

图5示出了根据本申请的一个实施例的区块链的异步落账装置的框图，该区块链的异步落账装置可以设置在区块链网络中的任意一个共识节点内。

参照图5所示，根据本申请的一个实施例的区块链的异步落账装置500，包括：第一处理单元502、第二处理单元504和持久化单元506。

其中，第一处理单元502配置为在区块达成共识之后，将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列中；第二处理单元504配置为在将所述各项数据分别写入对应的消息队列中之后，开启下一轮的区块共识过程；持久化单元506配置为分别从所述各项数据对应的消息队列中读取数据进行区块链账本的持久化处理。

在本申请的一些实施例中，基于前述方案，所述第一处理单元还配置为：将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列及缓存中。

在本申请的一些实施例中，基于前述方案，第一处理单元502还配置为：在所述区块达成共识之后，将所述区块及对应的区块执行结果写入日志中；针对所述日志中包含的目标区块，在对所述目标区块对应的各项数据进行持久化

处理之后，从所述日志中删除所述目标区块及所述目标区块的区块执行结果。

在本申请的一些实施例中，基于前述方案，第一处理单元502还配置为：在区块链节点重启后，获取所述日志中所包含的区块及区块执行结果；根据从所述日志中获取到的区块及区块执行结果进行持久化处理。

- 5 在本申请的一些实施例中，基于前述方案，第一处理单元502配置为：在将所述区块及对应的区块执行结果成功写入所述日志之后，将所述区块所对应区块信息中所包含的各项数据分别写入所述各项数据对应的消息队列中。

- 10 在本申请的一些实施例中，基于前述方案，第一处理单元502还配置为：若将所述区块及对应的区块执行结果写入所述日志失败，则进行重试处理，当重试次数达到设定次数后仍未成功写入所述日志，停止对所述区块进行落账处理并发出告警通知消息。

在本申请的一些实施例中，基于前述方案，第二处理单元504还配置为：在对目标区块对应的各项数据进行持久化处理之后，从所述缓存中删除所述目标区块对应的各项数据。

- 15 在本申请的一些实施例中，基于前述方案，第一处理单元502配置为：将所述区块信息中所包含的区块头写入区块头对应的队列中及缓存中，将所述区块信息中所包含的交易数据写入交易数据对应的队列中及缓存中，将所述区块信息中所包含的写集合写入写集合对应的队列中及缓存中，其中，所述交易数据包括交易信息列表及交易执行结果。

- 20 在本申请的一些实施例中，基于前述方案，所述区块信息中包含的各项数据包括：区块头、交易数据及写集合；第一处理单元502配置为：以区块高度作为关键字、区块头数据作为值，将所述区块信息中包含的区块头写入所述区块头对应的缓存中；以交易哈希作为关键字、交易数据作为值，将所述区块信息中包含的交易数据写入所述交易数据对应的缓存中；以表名和索引作为关键字、写数据作为值，将所述区块信息中包含的写集合写入所述写集合对应的缓存中。

在本申请的一些实施例中，基于前述方案，所述持久化单元506配置为：对所述各项数据对应的消息队列进行监听目标消息队列为所述各项数据对应的消息队列中的一个消息队列；若监听到所述目标消息队列中包含有数据，则

从所述目标消息队列中依次读取数据进行持久化处理。

在本申请的一些实施例中，基于前述方案，所述持久化单元506配置为：
调用存储层接口对从所述目标消息队列中依次读取出的数据进行持久化处理；
若持久化处理失败，则重新调用所述存储层接口对从所述目标消息队列中读取
5 出的数据进行持久化处理，直到持久化处理成功。

在本申请的一些实施例中，基于前述方案，所述持久化单元506配置为：
启动针对各个所述消息队列的监听任务；通过针对各个所述消息队列的监听任
务，对所述各项数据对应的消息队列进行监听。

在本申请的一些实施例中，基于前述方案，第二处理单元504还配置为：
10 在对所述区块信息完成持久化处理之前，通过所述缓存中存储的数据对业务请
求进行响应。

图6示出了适于用来实现本申请实施例的电子设备的计算机系统的结构示意图。

需要说明的是，图6示出的电子设备的计算机系统600仅是一个示例，不应
15 对本申请实施例的功能和使用范围带来任何限制。

如图6所示，计算机系统600包括中央处理单元（Central Processing Unit，
CPU）601，其可以根据存储在只读存储器（Read-Only Memory，ROM）602
中的程序或者从存储部分608加载到随机访问存储器（Random Access Memory，
RAM）603中的程序而执行各种适当的动作和处理，例如执行上述实施例中所述
20 的方法。在RAM 603中，还存储有系统操作所需的各种程序和数据。CPU 601、
ROM 602以及RAM 603通过总线604彼此相连。输入/输出（Input /Output，I/O）
接口605也连接至总线604。

以下部件连接至I/O接口605：包括键盘、鼠标等的输入部分606；包括诸
如阴极射线管（Cathode Ray Tube，CRT）、液晶显示器（Liquid Crystal Display，
25 LCD）等以及扬声器等的输出部分607；包括硬盘等的存储部分608；以及包括
诸如LAN（Local Area Network，局域网）卡、调制解调器等的网络接口卡的
通信部分609。通信部分609经由诸如因特网的网络执行通信处理。驱动器610
也根据需要连接至I/O接口605。可拆卸介质611，诸如磁盘、光盘、磁光盘、
半导体存储器等等，根据需要安装在驱动器610上，以便于从其上读出的计算

机程序根据需要被安装入存储部分608。

特别地，根据本申请的实施例，上文参考流程图描述的过程可以被实现为计算机软件程序。例如，本申请的实施例包括一种计算机程序产品，其包括承载在计算机可读介质上的计算机程序，该计算机程序包含用于执行流程图所示的方法的计算机程序。在这样的实施例中，该计算机程序可以通过通信部分609从网络上被下载和安装，和/或从可拆卸介质611被安装。在该计算机程序被中央处理单元（CPU）601执行时，执行本申请的系统中限定的各种功能。

需要说明的是，本申请实施例所示的计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质或者是上述两者的任意组合。计算机可读存储介质例如可以是——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件，或者任意以上的组合。计算机可读存储介质的更具体的例子可以包括但不限于：具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机访问存储器（RAM）、只读存储器（ROM）、可擦式可编程只读存储器（Erasable Programmable Read Only Memory, EPROM）、闪存、光纤、便携式紧凑磁盘只读存储器（Compact Disc Read-Only Memory, CD-ROM）、光存储器件、磁存储器件、或者上述的任意合适的组合。在本申请中，计算机可读存储介质可以是任何包含或存储程序的有形介质，该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。而在本申请中，计算机可读的信号介质可以包括在基带中或者作为载波一部分传播的数据信号，其中承载了计算机可读的计算机程序。这种传播的数据信号可以采用多种形式，包括但不限于电磁信号、光信号或上述的任意合适的组合。计算机可读的信号介质还可以是计算机可读存储介质以外的任何计算机可读介质，该计算机可读介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。计算机可读介质上包含的计算机程序可以用任何适当的介质传输，包括但不限于：无线、有线等等，或者上述的任意合适的组合。

附图中的流程图和框图，图示了按照本申请各种实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。其中，流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分，上述模块、程序段、或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应

当注意，在有些作为替换的实现中，方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如，两个接连地表示的方框实际上可以基本并行地执行，它们有时也可以按相反的顺序执行，这依所涉及的功能而定。也要注意的，框图或流程图中的每个方框、以及框图或流程图中的方框的组合，可以用执行规定的功能或操作的专用的基于硬件的系统来实现，或者可以用专用硬件与计算机指令的组合来实现。

描述于本申请实施例中所涉及到的单元可以通过软件的方式实现，也可以通过硬件的方式来实现，所描述的单元也可以设置在处理器中。其中，这些单元的名称在某种情况下并不构成对该单元本身的限定。

作为另一方面，本申请还提供了一种计算机可读介质，该计算机可读介质可以是上述实施例中描述的电子设备中所包含的；也可以是单独存在，而未装配入该电子设备中。上述计算机可读介质承载有一个或者多个程序，当上述一个或者多个程序被一个该电子设备执行时，使得该电子设备实现上述实施例中所述的方法。

应当注意，尽管在上文详细描述中提及了用于动作执行的设备的若干模块或者单元，但是这种划分并非强制性的。实际上，根据本申请的实施方式，上文描述的两个或更多模块或者单元的特征和功能可以在一个模块或者单元中具体化。反之，上文描述的一个模块或者单元的特征和功能可以进一步划分为由多个模块或者单元来具体化。

通过以上的实施方式的描述，本领域的技术人员易于理解，这里描述的示例实施方式可以通过软件实现，也可以通过软件结合必要的硬件的方式来实现。因此，根据本申请实施方式的技术方案可以以软件产品的形式体现出来，该软件产品可以存储在一个非易失性存储介质（可以是CD-ROM，U盘，移动硬盘等）中或网络上，包括若干指令以使得一台计算设备（可以是个人计算机、服务器、触控终端、或者网络设备等）执行根据本申请实施方式的方法。

本领域技术人员在考虑说明书及实践这里公开的实施方式后，将容易想到本申请的其它实施方案。本申请旨在涵盖本申请的任何变型、用途或者适应性变化，这些变型、用途或者适应性变化遵循本申请的一般性原理并包括本申请未公开的本技术领域中的公知常识或惯用技术手段。

— 18 —

应当理解的是, 本申请并不局限于上面已经描述并在附图中示出的精确结构, 并且可以在不脱离其范围进行各种修改和改变。本申请的范围仅由所附的权利要求来限制。

权 利 要 求

1. 一种区块链的异步落账方法, 所述方法由所述区块链中的共识节点执行, 所述方法包括:

在区块达成共识之后, 将所述区块所对应区块信息中包含的各项数据分别
5 写入所述各项数据对应的消息队列中;

在将所述各项数据分别写入对应的消息队列中之后, 开启下一轮的区块共识过程;

分别从所述各项数据对应的消息队列中读取数据进行区块链账本的持久化处理。

10 2. 根据权利要求 1 所述的方法, 其中所述将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列中, 包括:

将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列及缓存中。

15 3. 根据权利要求 1 所述的异步落账方法, 其中所述异步落账方法还包括:
在所述区块达成共识之后, 将所述区块及对应的区块执行结果写入日志中;
针对所述日志中包含的目标区块, 在对所述目标区块对应的各项数据进行持久化处理之后, 从所述日志中删除所述目标区块及所述目标区块的区块执行结果。

20 4. 根据权利要求 3 所述的异步落账方法, 其中所述异步落账方法还包括:
在区块链节点重启后, 获取所述日志中所包含的区块及区块执行结果;
根据从所述日志中获取到的区块及区块执行结果进行持久化处理。

5. 根据权利要求 3 所述的异步落账方法, 其中所述将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列中, 包括:
在将所述区块及对应的区块执行结果成功写入所述日志之后, 将所述区块
25 所对应区块信息中所包含的各项数据分别写入所述各项数据对应的消息队列中。

6. 根据权利要求 5 所述的异步落账方法, 其中所述异步落账方法还包括:
若将所述区块及对应的区块执行结果写入所述日志失败, 则进行重试处理,
当重试次数达到设定次数后仍未成功写入所述日志, 停止对所述区块进行落账

处理并发出告警通知消息。

7. 根据权利要求 2 所述的异步落账方法, 其中所述异步落账方法还包括:

在对目标区块对应的各项数据进行持久化处理之后, 从所述缓存中删除所述目标区块对应的各项数据。

5 8. 根据权利要求 2 所述的异步落账方法, 其中所述将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列及缓存中, 包括:

10 将所述区块信息中所包含的区块头写入区块头对应的队列中及缓存中, 将所述区块信息中所包含的交易数据写入交易数据对应的队列中及缓存中, 将所述区块信息中所包含的写集合写入写集合对应的队列中及缓存中, 其中, 所述交易数据包括交易信息列表及交易执行结果。

9. 根据权利要求 2 所述的异步落账方法, 其中所述区块信息中包含的各项数据包括: 区块头、交易数据及写集合;

15 所述将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的缓存中, 包括:

以区块高度作为关键字、区块头数据作为值, 将所述区块信息中包含的区块头写入所述区块头对应的缓存中;

以交易哈希作为关键字、交易数据作为值, 将所述区块信息中包含的交易数据写入所述交易数据对应的缓存中;

20 以表名和索引作为关键字、写数据作为值, 将所述区块信息中包含的写集合写入所述写集合对应的缓存中。

10. 根据权利要求 1 所述的异步落账方法, 其中所述分别从所述各项数据对应的消息队列中读取数据进行区块链账本的持久化处理, 包括:

25 对所述各项数据对应的消息队列进行监听, 目标消息队列为所述各项数据对应的消息队列中的一个消息队列;

若监听到所述目标消息队列中包含有数据, 则从所述目标消息队列中依次读取数据进行持久化处理。

11. 根据权利要求 10 所述的异步落账方法, 其中所述从所述目标消息队列中依次读取数据进行持久化处理, 包括:

调用存储层接口对从所述目标消息队列中依次读取出的数据进行持久化处理;

若持久化处理失败,则重新调用所述存储层接口对从所述目标消息队列中读取出的数据进行持久化处理,直到持久化处理成功。

- 5 12. 根据权利要求 10 所述的异步落账方法,其中所述对所述各项数据对应的消息队列进行监听,包括:

启动针对各个所述消息队列的监听任务;

通过针对各个所述消息队列的监听任务,对所述各项数据对应的消息队列进行监听。

- 10 13. 根据权利要求 2 所述的异步落账方法,其中所述异步落账方法还包括:
在对所述区块信息完成持久化处理之前,通过所述缓存中存储的数据对业务请求进行响应。

14. 一种区块链的异步落账装置,包括:

- 15 第一处理单元,配置为在区块达成共识之后,将所述区块所对应区块信息中包含的各项数据分别写入所述各项数据对应的消息队列中;

第二处理单元,配置为在将所述各项数据分别写入所述各项数据对应的消息队列中之后,开启下一轮的区块共识过程;

持久化单元,配置为分别从所述各项数据对应的消息队列中读取数据进行区块链账本的持久化处理。

- 20 15. 一种计算机可读介质,其上存储有计算机程序,所述计算机程序被处理器执行时实现如权利要求 1 至 13 中任一项所述的区块链的异步落账方法。

16. 一种电子设备,包括:

一个或多个处理器;

- 25 存储装置,用于存储一个或多个程序,当所述一个或多个程序被所述一个或多个处理器执行时,使得所述一个或多个处理器实现如权利要求 1 至 13 中任一项所述的区块链的异步落账方法。

17. 一种包括指令的计算机程序产品,当其在计算机上运行时,使得所述计算机执行权利要求 1 至 13 中任一项所述的区块链的异步落账方法。