

基于分布式存储的数据处理方法、装置、设备以及介质

本申请要求于 2021 年 02 月 19 日提交中国专利局、申请号为 202110188344X、申请名称为“基于分布式存储的数据处理方法、装置、设备以及介质”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

本申请涉及数据存储技术领域，尤其涉及区块链网络中的数据存储技术。

背景技术

10 目前，在区块链系统中，参与进行共识的共识节点进行数据存储时，通常会使用本地存储作为数据存储介质，比如，在共识节点 A 通过共识层对区块 N 进行区块共识的过程中，可以调用执行层执行该区块 N 中的各个交易，达成区块共识后，将各个交易的交易执行结果发送给存储层，以通过存储层将该区块 N 的区块信息（例如，区块头、交易列表、交易执行结果等）写入本地存储。通常情况下，共识节点中的本地存储和存储层为两个彼此独立
15 的硬盘空间，因此使用共识节点中的本地存储进行数据存储时，会受整个硬盘空间的大小的限制，而导致本地存储的存储量有限。

此外，在其他区块链系统中，还可以允许共识节点使用独立的数据库进行数据存储，然而，对于此类区块链系统而言，采用独立的数据库进行数据存储时，通常会因区块链系统所维护的区块链的块链结构的限制，而导致存储性能下降；比如，存储层将某个区块成功存储至相应的数据库后，才可以
20 继续通知共识层执行下一区块，这意味着在数据存储的过程中，需要按照每个区块的生成时间戳串行地存储数据，而这势必会降低区块链系统中数据的存储效率。

发明内容

25 本申请实施例提供一种基于分布式存储的数据处理方法、装置、设备以及介质，可以在提升数据存储量的同时，提升数据的存储效率。

本申请实施例一方面提供了一种基于分布式存储的数据处理方法，由区块链节点执行，包括：

30 将第一区块的第一块数据写入本地缓存和本地存储后，检测与本地缓存相关联的存储缓冲池的第一区块处理空间，得到第一空间检测结果；存储缓冲池中包括 N 个待存块数据，N 为非负整数；

若第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值，则基于 N 个待存块数据，将第一块数据写入存储缓冲池；

35 检测与存储缓冲池相关联的存储处理池的第二区块处理空间，得到第二

空间检测结果；存储处理池用于存储与分布式数据库相关联的 M 个待反馈块数据， M 为非负整数；

- 5 若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值，则基于待反馈块数据的数量和反馈区块处理阈值，在存储缓冲池中确定待合并块数据，对待合并块数据和 M 个待反馈块数据进行重叠检测，将重叠检测后保留的数据写入分布式数据库。

本申请实施例一方面提供了一种基于分布式存储的数据处理装置，包括：

- 10 第一检测模块，用于将第一区块的第一块数据写入本地缓存和本地存储后，检测与本地缓存相关联的存储缓冲池的第一区块处理空间，得到第一空间检测结果；存储缓冲池中包括 N 个待存块数据， N 为非负整数；

第一写入模块，用于若第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值，则基于 N 个待存块数据，将第一块数据写入存储缓冲池；

- 15 第二检测模块，用于检测与存储缓冲池相关联的存储处理池的第二区块处理空间，得到第二空间检测结果；存储处理池用于存储与分布式数据库相关联的 M 个待反馈块数据， M 为非负整数；

- 20 第二写入模块，用于若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值，则基于待反馈块数据的数量和反馈区块处理阈值，在存储缓冲池中确定待合并块数据，对待合并块数据和 M 个待反馈块数据进行重叠检测，将重叠检测后保留的数据写入分布式数据库。

本申请实施例一方面提供了一种计算机设备，包括存储器和处理器，存储器与处理器相连，存储器用于存储计算机程序，处理器用于调用计算机程序，以使得该计算机设备执行本申请实施例提供的方法。

- 25 本申请实施例一方面提供了一种计算机可读存储介质，计算机可读存储介质中存储有计算机程序，计算机程序适于由处理器加载并执行，以使得具有处理器的计算机设备执行本申请实施例提供的方法。

- 30 本申请实施例一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本申请实施例提供的方法。

- 35 本申请实施例在使用分布式数据库进行数据存储的过程中，可以先在将第一区块的第一块数据（即当前已经执行完毕的区块的块数据）写入本地缓存和本地存储（注意，这里的本地缓存和本地存储为在区块链节点的存储层中开辟出的两个存储空间）。然后，本申请实施例可以检测与本地缓存相关联的存储缓冲池的第一区块处理空间，得到第一空间检测结果；位于存储层中的存储缓冲池当前可以包括 N 个待存块数据，这里的 N 可以为非负整数。

进一步的，若第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值（例如，10个），则该区块链节点可以基于N个待存块数据将第一块数据写入存储缓冲池。进一步的，区块链节点可以检测与存储缓冲池相关联的存储处理池的第二区块处理空间，
5 得到第二空间检测结果；这里的存储处理池可以用于存储与分布式数据库相关联的M个待反馈块数据，这里的M可以为非负整数。若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值，则区块链节点可以基于待反馈块数据的数量和反馈区块处理阈值，从存储缓冲池中确定待合并块数据，进而可以对待合并块数据和
10 M个待反馈块数据进行重叠检测，以便后续将重叠检测后保留的数据写入分布式数据库。可见，本申请实施例在进行数据存储的过程中，为了提升数据的存储量，采用分布式数据库来替换本地存储。此外，为了在使用分布式数据库进行数据存储的过程中提升数据的读写效率，本申请实施例还对存储层进行了扩展，即在存储层中扩展了本地缓存、本地存储、存储缓冲池以及存
15 储处理池等，重构得到一个全新的存储层。这样，在使用该存储层得到存储至分布式数据库的数据的过程中，不仅可以实现多个块的并行存储，还可以从根源上突破整个区块链系统维护的块链结构的限制。换言之，本申请实施例中的区块链节点在将多个块并行存储至分布式数据库的过程中，可以提升数据存储量和数据存储效率。

20 附图说明

图1是本申请实施例提供的一种网络架构的示意图；

图2是本申请实施例提供的一种基于分布式存储进行数据存储的场景示意图；

25 图3是本申请提供的一种基于分布式存储的数据处理方法的流程示意图；

图4是本申请实施例提供的一种执行第一区块和执行第二区块的场景示意图；

图5是本申请实施例提供的一种基于分布式存储的数据处理方法的流程示意图；

30 图6是本申请实施例提供的一种读取目标数据的场景示意图；

图7是本申请实施例提供的一种存储层的场景示意图；

图8是本申请实施例提供的一种数据打包单元的场景示意图；

图9是本申请实施例提供的一种数据交互图；

35 图10是本申请提供的一种基于分布式存储的数据处理装置的结构示意图；

图11是本申请实施例提供的一种计算机设备的结构示意图。

具体实施方式

下面将结合本申请实施例中的附图，对本申请实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本申请一部分实施例，而不是全部的实施例。基于本申请中的实施例，本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例，都属于本申请保护的范围。

请参见图 1，图 1 是本申请实施例提供的一种网络架构的示意图。图 1 所示的网络架构可以应用于区块链系统，该区块链系统可以是由多个节点通过网络通信的形式连接形成的分布式系统。其中，该区块链系统可以包括但不限于联盟链对应的区块链系统。

其中，区块链是一种分布式数据存储、点对点传输、共识机制、以及加密算法等计算机技术的新型应用模式，主要用于按时间顺序对数据进行整理，并加密成账本，使其不可被篡改和伪造，同时可进行数据的验证、存储和更新。区块链本质上是一个去中心化的数据库，该数据库中的每个节点均存储一条相同的区块链，区块链网络将节点区分为核心节点和轻节点，其中，核心节点可以负责区块链全网的共识，也就是说，核心节点可以为区块链网络中的共识节点。区块链网络中交易数据被写入账本（例如，本地账本）的过程可以为，客户端发送交易数据至轻节点，随后该交易数据以接力棒的方式在区块链网络中的轻节点之间传递，直到共识节点收到该交易数据，共识节点再将该交易数据打包进区块，以便后续可以与其他共识节点之间进行共识，在共识通过后，可以通过存储层将携带该交易数据的多个区块并行写入分布式数据库，这样可以从根源上突破区块链的区块链结构的限制，进而有效地提升数据存储的存储效率。

其中，区块链系统中可以包括智能合约，该智能合约在区块链系统中可以理解为是一种由区块链各节点（包括共识节点）执行的代码，通过该智能合约可以执行任意逻辑并得到结果。比如，用户可以通过客户端发起交易业务请求，以调用区块链上已经部署的智能合约，随后，区块链上的数据节点或轻节点可以将该交易业务请求发送至共识节点，以调用各个共识节点中运行的智能合约执行该用户所请求的交易业务。应当理解，区块链中可以包括一个或多个智能合约，这些智能合约可以通过合约调用地址、合约标识号（Identity document, ID）或合约名称来区分，而在客户端发起的交易业务请求中，可以携带智能合约的合约调用地址或者合约标识号或合约名称，以指定需要运行的智能合约。若客户端所指定的智能合约为需要读取数据的合约（即业务合约），则各个共识节点可以通过访问在存储层中创建的本地缓存来进行数据的读取，最后各个共识节点会互相验证各交易执行结果是否一致（也就是进行共识），若一致，则可以将交易执行结果存入各自的本地缓存和本地存储中，并可以将上述交易业务的交易执行结果返回至客户端。注意，这里的本地缓存为在存储层中创建的系统内存，这里的本地存储为在存储层

中创建的用于进行数据存储的硬盘空间。这样，当某个共识节点出现宕机（即down机）或者系统故障时，不会因为系统内存中的数据消失而发生无法进行数据读取的情况，即该共识节点可以通过在该存储层中创建的本地存储来进行数据的读取。

5 应当理解，图1所示的网络架构可以包括核心节点（即共识节点）集群，轻节点集群以及用户终端集群。其中，该核心节点集群可以包括多个核心节点，这里的多个核心节点具体可以包括图1所示的节点10a、节点10b、节点10c、节点10d。如图1所示，节点10a、节点10b、节点10c和节点10d之间可以通过网络连接，构成图1所示的共识网络100a。可以理解的是，在共识网络100a中，节点10a、节点10b、节点10c和节点10d之间均能通过网络连接进行数据交互。此外，用户终端集群可以包括多个用户终端，这里的多个用户终端具体可以包括图1所示的用户终端3000a、用户终端3000b、...、用户终端3000c。如图1所示，用户终端3000a、用户终端3000b、...、用户终端3000c可以分别与节点10c进行网络连接，以便通过该节点10c之间的网络连接进行数据交互。另外，轻节点集群可以包括多个轻节点，这里的多个轻节点具体可以包括图1所示的服务器4000a和服务器4000b。如图1所示，服务器4000a和服务器4000b可以分别与节点10c进行网络连接，以便通过该节点10c之间的网络连接进行数据交互。

20 本申请实施例可以将共识网络100a中的每个核心节点（例如，节点10a、节点10b、节点10c、节点10d）统称为区块链节点。应当理解，这些区块链节点可以用于维护同一区块链（比如，图1所示的区块链10e），该共识网络100a中的任意两个区块链节点之间可以形成点对点（P2P，Peer To Peer）网络，该点对点网络可以采用P2P协议，该P2P协议是一个运行在传输控制协议（TCP，Transmission Control Protocol）协议之上的应用层协议。在分布式系统中，任何设备如服务器、终端等都可以加入区块链网络，而成为区块链节点，每个区块链节点均可以包括硬件层、中间层、操作系统层和应用层。

25 本申请实施例可以为接入该区块链网络的任意一个角色（例如，任意一个个人用户、任意一个企业、任意一个机构等实体对象）绑定一个区块链节点，这些区块链节点构成的区块链网络可以统称为联盟链网络。所以，图1所示的节点10a、节点10b、节点10c、节点10d与需要接入该联盟链网络中的相应角色（即相应业务场景下的实体对象）之间存在一一对应关系。这里的业务场景可以包括电子票据场景、社交场景、赊购场景、信贷场景等。此时，相应业务场景下的目标业务具体可以包括电子票据业务、社交业务、赊购业务、信贷业务等，这里不对相应业务场景下的具体业务进行一一列举。

30 由于每个实体对象均可以对应一个区块链节点，因此，本申请实施例可以以实体对象为上述企业用户（即前述企业）为例，此时，与每个企业用户相关联的区块链节点可以为同一区块链节点（例如，上述图1所示的节点10c

可以与多个企业用户各自对应的用户终端进行数据交互)。比如,在区块链电子票据系统中,可以将每个开票企业对应的电子票据业务(比如,注册业务、开票业务、票据转移业务等)统称为一笔交易业务。其中,开票企业 A 可以通过图 1 所示的用户终端 3000a 与图 1 所示的节点 10c 进行数据交互,以完成相应的交易;以此类推,开票企业 B 可以通过图 1 所示的用户终端 3000b 与图 1 所示的节点 10c 进行数据交互,以完成相应的交易;开票企业 C 可以通过图 1 所示的用户终端 3000c 与图 1 所示的节点 10c 进行数据交互,以完成相应的交易。

本申请实施例可以将上述针对电子票据业务发送交易业务请求的实体对象(例如,开票企业 A、开票企业 B、...、开票企业 C)统称为目标用户,并可以将接收目标用户发送的交易业务请求的区块链节点统称为轻节点,还可以将参与对该交易业务请求进行共识的区块链节点统称为核心节点。

可选的,在共识网络 100a 中,由于节点 10c 可以与其存在网络连接(也可以称之为会话连接)的其他区块链节点之间进行数据同步,即节点 10c 可以从其他区块链节点上同步相应的业务数据信息(例如,可以从其他区块链节点上同步其他企业用户发送的交易业务请求等),此时,与每个企业用户相关联的核心节点可以为不同的区块链节点。比如,开票企业 A 也可以通过图 1 所示的用户终端 3000a 与图 1 所示的节点 10c 进行数据交互;开票企业 B 可以通过图 1 所示的用户终端 3000b 与图 1 所示的节点 10b 进行数据交互;开票企业 C 也可以通过图 1 所示的用户终端 3000c 与图 1 所示的节点 10a 进行数据交互。由于区块链网络中的这些节点均会维护同一区块链,因此,通过将不同用户终端发送的交易业务请求随机分配给与区块链 10e 相关联的区块链节点,可以有效地均衡区块链网络中的网络负载,从而提高相应业务对应的业务数据的处理效率。

又比如,在信贷系统中,可以将目标用户(例如,上述用户终端 3000a 对应的用户)请求的信贷业务理解为另一笔交易业务。其中,目标用户可以通过图 1 所示的用户终端 3000a 与图 1 所示的节点 10c 进行数据交互,以向图 1 所示的节点 10c 发送针对相应业务的交易业务请求。即在该信贷系统中,可以为该接入该信贷系统中的每个用户配置一个区块链节点,以接收相应用户发送的交易业务请求。应当理解,本申请实施例中的每个用户也可以统称为实体对象,比如,请求前述信贷业务的实体对象可以为前述个人用户、企业用户等。

轻节点接收到交易业务请求后,可以将该目标用户发起的交易业务请求转发给核心节点,以通过核心节点对该目标用户发起的交易业务请求进行合法性验证。核心节点可以在交易业务请求通过合法性验证后,将该第一用户请求的交易业务添加至交易池,以便后续将与该交易业务相关联的交易数据打包成区块,使共识网络 100a 中的共识节点进行区块共识,并在区块共识通

过后，将该区块的块数据写入至本地缓存和本地存储，以便于后续可以基于上述分布式存储实现多个区块的块数据的并行存储。

为便于理解，进一步的，请参见图 2，图 2 是本申请实施例提供的一种基于分布式存储进行数据存储的场景示意图。图 2 所示的区块链节点可以为上述图 1 对应的实施例中的任意一个核心节点。该区块链节点可以通过共识层 100、执行层 200、存储层 300 以及分布式存储 400，实现多个区块（简称多个块）的并行存储。

其中，共识层 100 的主要功能可以包括区块打包功能、区块广播功能和区块共识功能；应当理解，该共识层 100 在通过区块共识功能进行区块共识的过程中，可以调用图 2 所示的执行层 200 执行块内的各个交易业务，以对当前待共识的区块中的各个交易业务进行交易验证，进而在交易验证成功后，确定达成对该区块 1 的区块共识。此时，如图 2 所示，该区块链节点可以将由执行层 200 执行完的各交易业务的交易执行结果统称为该区块 1 的块数据，进而可以将该区块 1 的块数据发送给图 2 所示的存储层。

本申请实施例可以将执行层 200 当前接收到的由共识层 100 发送的区块 N 统称为第一区块，并可以将该执行层 200 执行完毕的第一区块的块数据统称为第一区块的第一块数据。应当理解，该第一区块的第一块数据可以包括但不限于第一区块中每个交易业务的交易执行结果，比如，该第一区块的第一块数据还可以包括该第一区块的区块头信息（即第一区块头信息）；可选的，该第一区块中的交易业务的数量可以为 K1 个，此时，该第一区块的第一块数据还可以包括与该 K1 个交易业务相关联的交易列表，这里的 K1 可以为正整数。

其中，执行层 200 在执行区块中的某个交易业务时，可以调用合约虚拟机（简称为虚拟机）中的智能合约，执行对应交易业务、并可以在执行完区块中的每个交易业务时，将各个交易执行结果发送给图 3 所示的存储层 300。

其中，存储层 300 可以负责将前述第一区块的区块信息（如区块头，交易列表、执行结果等块数据）存入至图 2 所示的分布式存储 400。这里，该分布式存储 400 作为数据的存储介质，可以独立于该区块链节点集成在其他分布式服务器上。如图 2 所示，存储层 300 中可以包括图 2 所示的灾备模块 310、本地缓存 320 和持久化模块 330。这里的本地缓存 320 可以为该区块链节点中创建的本地缓存模块。该本地缓存 320 可以为区块链节点的系统内存。

如图 2 所示，该灾备模块 310 可以位于该区块链节点的存储硬盘中，图 2 所示的本地存储 312 可以用于存储（即备份）与本地缓存 320 相同的块数据，这样，当该区块链节点出现宕机或者系统故障时，就不会因为系统内存中缓存的块数据消失，而影响到区块中的其他交易业务的执行，此时，该区块链节点可以使用该灾备模块 310，从节点的存储硬盘中读取执行区块中的其他交易业务所需的目标数据。本地存储 312 可以为该灾备模块 310 中的具

有持久化能力的本地存储单元。

如图 2 所示，区块链节点在将第一区块的第一块数据写入到图 2 所示的本地存储 312 和本地缓存 320 时，可以通知图 2 所示的共识层 100 处理该第一区块的下一区块。可以理解的是，本申请实施例可以将第一区块的下一区块统称为新的第一区块，为便于对共识层 100 在不同时刻处理的区别进行区别，本申请实施例可以将当前时刻的下一时刻处理的区块（即前述新的第一区块）称为第二区块，可以将第二区块的块数据统称为第二块数据。

其中，如图 2 所示，存储层 300 中的持久化模块 330 可以包括图 2 所示的存储缓冲池 331 和存储处理池 332。存储缓冲池 331 可以用于存放执行层 200 发送的已执行完毕的第一区块的第一块数据，该第一块数据可以与该存储缓冲池 331 当前所存放的数据（例如，N 个待存块数据）进行数据合并，从而减少因相同的数据重复存储而造成的存储缓冲池 331 的存储空间的浪费。该存储缓冲池 331 的区块处理空间（即第一区块处理空间）的大小可以根据实际业务需求进行自适应的配置。

此外，在本申请实施例中，区块链节点将第一区块的第一块数据写入到本地缓存 320 和本地存储 312 后，还可以通过第一刷新线程对存储缓冲池的第一区块处理空间进行检测，若根据空间检测结果（即第一空间检测结果）确定该存储缓冲池 331 具备处理当前区块的块数据的空间时，允许基于前述 N 个待存块数据将第一块数据写入至该存储缓冲池 331。应当理解，通过配置该存储缓冲池 331 的空间大小，可以从根源上解决因区块执行过快而造成的数据读取错误的问题。比如，若不合理配置空间大小，那么，区块执行过快时，就会使区块链节点将本地缓存 320 中还没来得及存储至分布式数据库 400 的数据（例如，一些存入时间比较久的数据）清除掉，进而造成数据读取错误的现象。所以，区块链节点在将第一块数据写入存储缓冲池 331 之后，可以通过前述第一刷新线程对该存储缓冲池 331 的第一区块处理空间进行缓冲刷新，这样，该区块链节点可以在刷新后的第一区块处理空间具备处理上述下一区块的空间时，允许进一步接收述执行层 200 发送的执行完的第二区块的第二块数据。

进一步的，当区块链节点将第一块数据写入存储缓冲池 331 之后，可以将该存储缓冲池 331 中的数据统称为待存数据，这样，当区块链节点通过第二刷新线程对图 2 所示的存储处理池 332 的区块处理空间（即第二区块处理空间）进行检测之后，若根据空间检测结果（即第二空间检测结果）确定该存储处理池 332 具备保持数据的空间时，允许从前述存储缓冲池 331 中获取相应数量的块数据（例如，一个或者多个块数据）作为待合并块数据，进而对获取到的待合并块数据和该存储处理池 332 中的 M 个待反馈块数据进行重叠检测，从而将重叠检测后的数据写入至图 2 所示的分布式存储 400 的分布式数据库。通过配置该存储处理池 332 的空间大小，可以在不过度消耗内存

的情况下，实现多个块的并行存储。

区块链节点将第一块数据写入存储缓冲池以及进行多个块的并行存储的具体过程，可以参见如下图 3 至图 9 对应的实施例。

进一步的，请参见图 3，图 3 是本申请提供的一种基于分布式存储的数据处理方法的流程示意图，如图 3 所示，方法可以由上述区块链网络中的区块链节点执行，比如，该区块链节点可以为上述图 1 所示的共识网络 100a 中任意一个核心节点。方法具体可以包括以下步骤 S101-步骤 S104。

步骤 S101，将第一区块的第一块数据写入本地缓存和本地存储后，检测与本地缓存相关联的存储缓冲池的第一区块处理空间，得到第一空间检测结果。

其中，存储缓冲池中包括 N 个待存块数据；N 为非负整数，且 N 用于表征第一区块处理空间的待存区块数量。

区块链节点在接收到由共识层（例如，上述图 2 所示的共识层 100）发送的第一区块时，可以获取该第一区块中的 K1 个交易业务，进而可以调用每个交易业务的业务合约执行对应的交易业务。其中，第一区块可以为共识层基于区块打包功能，在当前时刻（如 T1 时刻）打包得到的区块，该 T1 时刻打包得到的区块可以为上述区块 1。

当区块链节点通过执行层（例如，上述图 2 所示的执行层 200）执行完该区块 1（即第一区块）时，可以将该区块 1 的第一块数据写入至本地缓存（例如，上述图 2 所示的本地缓存 320）和本地存储（例如，上述图 2 所示的本地存储 312）。区块链节点将第一块数据写入到本地缓存和本地存储后，可以立刻通知共识层处理下一区块。进而，该共识层可以基于区块打包功能，进一步从交易池中获取 K2 个交易业务进行打包，以生成新的待共识的区块（例如，上述区块 2）。本申请实施例可以将共识层在 T2 时刻打包得到的区块统称为第二区块，这里的第二区块中的交易业务的数量（即 K2）与上述第一区块中的交易业务的数量（即 K1）可以相同，也可以不同，这里不对 K1 和 K2 的具体取值进行限定。上述 T1 时刻和 T2 时刻为两个不同的打包时刻，且 T2 时刻处于 T1 时刻之后。

为便于理解，进一步的，请参见图 4，图 4 是本申请实施例提供的一种执行第一区块和第二区块的场景示意图。其中，图 4 所示的第一区块可以为上述共识层在 T1 时刻打包得到的待验证的区块 1，该待验证的区块 1 可以包括 K1 个交易业务，这 K1 个交易业务具体可以包括图 4 所示的交易 11、交易 12、交易 13、...、交易 1k。应当理解，当区块链节点的执行层 H 获取到这 K1 个交易业务时，可以对这 K1 个交易业务中的每个交易业务进行交易校验，并在交易校验成功后，得到每个交易业务的交易执行结果。如图 4 所示，区块 1 中每个交易业务的交易执行结果可以为图 4 所示的第一交易执行结果，在该第一交易执行结果中，结果 21 可以为交易 11 对应交易执行结果，结果

22 可以为交易 12 对应交易执行结果，结果 23 可以为交易 13 对应交易执行结果，以此类推，结果 2k 可以为交易 1k 对应交易执行结果。

5 区块链节点完成交易校验后，可以生成校验成功指示信息，若该区块链节点的共识层接收到超过一半的其他共识节点发送的校验成功指示信息，则可以确定达成对该第一区块的区块共识，进而可以将达成共识的第一区块（即图 4 所示的区块 1）的第一块数据写入到本地缓存和本地存储。该区块链节点将该第一区块的第一块数据写入本地缓存和本地存储后，可以通过第一刷新线程异步对存储缓冲池（例如，上述图 2 所示的存储缓冲池 331）的区块处理空间（即上述第一区块处理空间）进行检测，以得到第一空间检测结果。
10 该第一空间检测结果可以用于表征该存储缓冲池在当前时刻（例如， $T1'$ 时刻，该 $T1'$ 时刻可以为上述 $T1$ 时刻之后的时刻，例如，上述 $T2$ 时刻）是否有处理当前区块的空间。

若该区块链节点通过第一刷新线程统计到存储缓冲池当前存储的待存块数据的数量为 N 个，则表示该存储缓冲池的第一区块处理空间能够缓冲的待存区块的数量（即待存区块数量）为 N 个。比如，若该第一空间检测结果指示该第一区块处理空间中待存块数据的数量（比如， $N=6$ ）未达到第一区块处理空间的缓冲区块处理阈值（例如， $N_{\max}=10$ ），则表明该存储缓冲池在当前时刻（例如，上述 $T1'$ 时刻）具有处理当前区块的空间，进而可以进一步执行下述步骤 S102。

20 可选的，若第一空间检测结果指示第一区块处理空间中待存块数据的数量（比如， $N=10$ ）已达到缓冲区块处理阈值（例如， $N_{\max}=10$ ），则可以通知上述执行层对第一块数据进行存储等待，这意味着此时该存储缓冲池没有处理当前块的空间，因此，上述执行层将需要进行等待，直到上述存储缓冲池具有处理当前区块的空间，才可以进一步执行下述步骤 S102。

25 步骤 S102，若第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值，则基于 N 个待存块数据，将第一块数据写入存储缓冲池。

具体的，若前述第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值，则区块链节点可以从存储缓冲池中获取 N 个待存块数据，进而可以在 N 个待存块数据中查找与第一块数据存在数据重叠的待存块数据；若在 N 个待存块数据中查找到与第一块数据存在数据重叠的待存块数据，则区块链节点可以将查找到的待存块数据作为重合块数据；进而，区块链节点可以将重合块数据替换为第一块数据，进而可以将替换后的 N 个待存块数据作为与第一块数据相关联的第一合并数据，
30 并可以将第一合并数据写入存储缓冲池。
35

应当理解，区块链节点在将第一块数据写入存储缓冲池的过程中，需要对该存储缓冲池当前存储的 N 个待存块数据进行合并处理，这样，可以减少

相同数据的重复存储的情况。

比如，区块链节点在将第一块数据写入存储缓冲池的过程中，若区块链节点在还未完全将整个区块的块数据写入存储缓冲池的情况下发生系统故障，则存储缓冲池中存储有该第一区块的部分块数据，这样，该区块链节点重启后，可以重新接收到上述执行层发送的区块 1（即前述第一区块），此时，区块链节点可以从存储层的本地存储中读取用于执行区块 1 中 K1 个交易业务的目标数据，进而可以得到 K1 个交易业务的交易执行结果。将这些重新生成的第一块数据写入到存储缓冲池时，可以在当前的 N 个待存块数据中查找与第一块数据存在数据重叠的待存块数据，进而，该区块链节点可以将查找到的待存块数据（即前述第一区块的部分块数据）作为重合块数据，并在 N 个待存块数据中将重合块数据替换为第一块数据，从而可以将替换后的 N 个待存块数据作为第一合并数据，并将第一合并数据写入存储缓冲池。由此可见，区块链节点可以在出现宕机或者系统故障时，进一步在 N 个待存块数据中，用重新生成的第一块数据替换掉前述系统故障或者宕机时缓冲的第一区块的部分块数据，以避免相同数据的重复写入。

可选的，若在 N 个待存块数据中未查找到与第一块数据存在数据重叠的待存块数据，则区块链节点可以将第一块数据和 N 个待存块数据作为与第一块数据相关联的第二合并数据，将第二合并数据写入存储缓冲池，且对第一区块处理空间中待存块数据的数量进行递增处理；递增处理后的第一区块处理空间的待存块数据的数量为 N+1。

可选的，区块链节点还可以在将第一块数据写入存储缓冲池时，进一步接收前述执行层发送的下一区块的块数据，这里的下一区块的块数据主要是指上述第二区块（例如，上述图 4 所示的区块 2）的第二块数据，并可以通过第一刷新线程继续对存储缓冲池的第一区块处理空间进行缓冲刷新，以得到缓冲刷新结果（比如，此时，该第一刷新线程可以统计到存储缓冲池在下一时刻（例如，T2' 时刻，即 T2' 时刻为上述 T1' 时刻的下一时刻）存储的待存块数据的数量为 N+1 个，此时，该区块链节点可以用该缓冲刷新结果更新第一空间检测结果，进而可以基于更新后的第一空间检测结果，判断该存储缓冲池在下一时刻（例如，T2' 时刻）是否具备处理下一区块的空间。若更新后的第一空间检测结果指示第一区块处理空间中待存块数据的数量仍未达到缓冲区块处理阈值，则可以进一步在下一时刻将第二块数据添加至第一块数据所在的存储缓冲池。

如上述图 4 所示的第二区块可以为上述共识层在前述 T2 时刻打包得到的待验证的区块 2，该待验证的区块 2 可以包括 K2 个交易业务，这 K2 个交易业务具体可以包括图 4 所示的交易 31、交易 32、交易 33、...、交易 3n。应当理解，区块链节点的执行层获取到这 K2 个交易业务后，可以对这 K2 个交易业务中的每个交易业务进行交易校验，并在交易校验成功时，得到每个交

易业务的交易执行结果。每个交易业务的交易执行结果可以为上述图 4 所示的第二交易执行结果，在该第二交易执行结果中，结果 41 可以为交易 31 对应交易执行结果，结果 42 可以为交易 32 对应交易执行结果，结果 43 可以为交易 33 对应交易执行结果，以此类推，结果 4n 可以为交易 3n 对应交易执行结果。

区块链节点完成对 K2 个交易业务的交易校验后，可以生成新的校验成功指示信息，这样，若该区块链节点的共识层接收到超过一半的其他共识节点发送的新的校验成功指示信息，则可以确定达成对该第二区块的区块共识，进而可以将达成共识的第二区块（即图 4 所示的区块 2）的第二块数据写入到本地缓存和本地存储，以便后续在检测到存储缓冲池具有处理下一区块的空间时，可以快速地将该第二区块的第二块数据写入到存储缓冲池。

步骤 S103，检测与存储缓冲池相关联的存储处理池的第二区块处理空间，得到第二空间检测结果。

其中，存储处理池用于存储与分布式数据库相关联的 M 个待反馈块数据；M 为非负整数，且 M 用于表征第二区块处理空间的待反馈区块数量。

区块链节点在执行完上述步骤 S103 之后，可以将存储缓冲池中的第一合并数据或者第二合并数据统称为待存数据，进而，可以在通过第二刷新线程，检测与上述存储缓冲池相关联的存储处理池的第二区块处理空间，得到第二空间检测结果。若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量（比如，M=10）达到反馈区块处理阈值（比如， $M_{\max}=10$ ），则该区块链节点可以通知上述存储缓冲池对待存数据进行存储等待，直到通过第二刷新线程检测得到的第二空间检测结果第二区块处理空间中待反馈块数据的数量未达到反馈区块处理阈值。这样，当区块链节点根据第二空间检测结果确定出前述待反馈区块数量未达到存储处理池的反馈区块处理阈值时，可以进一步执行下述步骤 S104。

若区块链节点在执行完上述步骤 S103 后，即通过第二刷新线程，检测与上述存储缓冲池相关联的存储处理池的第二区块处理空间之后，确定出第二空间检测结果所指示的第二区块处理空间中待反馈块数据的数量（比如，M=7）未达到反馈区块处理阈值（比如， $M_{\max}=10$ ），则可以直接执行下述步骤 S104。

步骤 S104，若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值，则基于待反馈块数据的数量和反馈区块处理阈值，在存储缓冲池中确定待合并块数据，对待合并块数据和 M 个待反馈块数据进行重叠检测，将重叠检测后保留的数据写入分布式数据库。

具体的，若存储缓冲池中的待存数据包括 N 个待存块数据和第一块数据，且第一块数据与 N 个待存块数据之间不存在数据重叠；则区块链节点可以在

第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值时，确定待反馈块数据的数量与反馈区块处理阈值之间的数量差值，进而可以基于数量差值以及存储缓冲池中的待存数据的区块高度，从N个待存块数据和第一块数据中确定待合并块数据，对待合并块数据和M个待反馈块数据进行重叠检测，得到重叠检测结果；进一步的，区块链节点可以在重叠检测结果指示M个待反馈块数据中不存在与待合并块数据相重叠的待反馈块数据时，将待合并块数据写入存储处理池，进而可以将存储处理池中的待合并块数据和前述M个待反馈块数据统称为重叠检测后保留的数据，并可以将重叠检测后保留的数据写入分布式数据库。注意，区块链节点在将多个块数据写入分布式数据库的过程中，是并行执行的，这样就可以从根源上突破现有区块存储的块链结构的限制，进而提升数据存储的效率。

可选的，区块链节点在将重叠检测后保留的数据写入分布式数据库的过程中，会源源不断地接收到分布式数据库返回的块数据中对应数据的存储完毕反馈信息。基于此，区块链节点接收到对应数据的存储完毕反馈信息，则间接反映在重叠检测后保留的数据中存在成功写入分布式数据库的数据，此时，区块链节点可以将成功写入分布式数据库的数据统所属的块数据称为目标块数据，且可以接收分布式数据库反馈的目标块数据的数据完成量；然后，区块链节点可以基于数据完成量，在块存储状态映射表中对目标块数据对应的待存数据量进行递减处理，直到待存数据量达到递减阈值（例如，0）时，确定已成功将目标块数据对应的完整区块写入分布式数据库。

存储处理池中存储的M个待反馈块数据为当前正在存储至分布式数据库，但还未接收到分布式数据库返回的存储完毕反馈信息的块数据。应当理解，前述重叠检测后保留的数据可以为多个块数据。这样，该区块链节点可以通过前述存储缓冲池和存储处理池，实现多个块的并行存储。基于此，在区块链节点将存储处理池中的这些块数据并行存储至分布式数据库的过程中，可以接收到分布式数据库返回的每个块数据中对应数据的存储完毕反馈信息，此时，区块链节点可以基于接收到的对应数据的存储完毕反馈信息（即前述目标块数据的数据完成量），在存储层中将块存储状态映射表中目标块数据对应区块的块高（即区块高度）的待存数据量进行递减处理，直到对应区块的块高的待存数据量达到递减阈值（例如，0）为止，此时，该区块链节点可以确定当前已经成功将目标块数据所对应块高的完整区块成功写入到分布式数据库。

由此可见，本申请实施例在进行数据存储的过程中，为了能够提升数据的存储量，采用分布式存储来替换以往的本地存储。此外，为了在使用分布式存储进行数据存储的过程中提升数据的读写效率，本申请实施例还在存储层中进行扩展，比如在存储层中扩展本地缓存、本地存储、存储缓冲池以及

存储处理池等，重构得到一个全新的存储层。这样，在使用该存储层得到存储至分布式数据库的数据的过程中，不仅可以实现多个块的并行存储，还可以从根源上突破整个区块链系统所维护的块链结构的限制。换言之，本申请实施例中的区块链节点（即参与进行共识的共识节点）在将多个块并行存储至分布式数据库的过程中，可以提升数据存储量，还可以提升数据存储效率。

进一步的，请参见图 5，图 5 是本申请实施例提供的一种基于分布式存储的数据处理方法的流程示意图。如图 5 所示，方法可以由区块链节点执行，该区块链节点可以为上述图 1 所示的共识网络 100a 中的任一个核心节点（例如，上述节点 10c）。该方法具体包括以下步骤 S201-S210：

步骤 S201，执行层接收到共识层发送的第一区块后，获取第一区块中的 K1 个交易业务。

其中，第一区块包括第一区块头信息、与 K1 个交易业务相关联的交易列表；K1 为正整数。

步骤 S202，针对 K1 个交易业务中的每个交易业务，获取用于执行该交易业务的业务合约，通过该交易业务的业务合约执行该交易业务，得到该交易业务的交易执行结果。

具体的，针对 K1 个交易业务中的每个交易业务，区块链节点可以基于该交易业务的合约调用地址，获取用于执行该交易业务的业务合约；进一步的，区块链节点可以通过该交易业务的业务合约，从本地缓存中查找与该交易业务的相关联的目标数据。若在本地缓存中查找到与该交易业务的相关联的目标数据，则区块链节点可以将本地缓存中查找到的目标数据作为该交易业务的目标读取数据，并可以基于该交易业务的目标读取数据执行所述该交易业务，得到所述该交易业务的交易执行结果。

为便于理解，进一步的，请参见图 6，图 6 是本申请实施例提供的一种读取目标数据的场景示意图。该区块链节点在执行第一区块中的某个交易业务（例如，交易业务 Tx1）时，可以通过虚拟机上运行的智能合约（即前述交易业务 Tx1 的业务合约）中的合约数据读取方法，从图 6 所示的本地数据库（例如，本地缓存对应的数据库 1，该数据库 1 可以为系统数据库）上获取用于执行该交易业务 Tx1 的目标数据。这里的目标数据可以根据上述业务场景的不同，自适应地进行配置，这里不对其进行限定。比如，在进行资产转移的过程中，这里的目标数据可以为请求进行资产转移的目标用户的账户余额等。

若在该区块链节点在执行该第一区块中的交易业务 Tx1 的下一交易业务（例如，交易业务 Tx2）的过程中，若出现宕机或者系统故障，则还可以在节点重启时，可以通过数据恢复线程重新接收前述第一区块，以基于本地存储中存储的数据（即备份数据）继续执行第一区块中的交易业务，此时，区块链节点也可以优先从图 6 的本地数据库（例如，本地存储对应的数据库 2，

该数据库 2 可以为硬盘数据库) 中读取需要执行相应交易业务的数据。正常情况下, 上述系统数据库可以视为节点中的主存储数据库, 且上述硬盘数据库可以视为节点中的备用存储数据库。由此可见, 本申请实施例可以在交易业务的执行过程中, 通过节点本地数据库中的主备切换, 提升故障时的容灾能力。此外, 本申请实施例通过在节点本地使用本地缓存, 可以在交易执行过程中, 有效地减轻使用分布式数据库带来的网络开销。

由此可见, 若区块链节点在本地缓存中查找到与每个交易业务的相关联的目标数据, 则该区块链节点可以将本地缓存中查找到的目标数据作为每个交易业务的目标读取数据; 进一步的, 区块链节点可以基于每个交易业务的目标读取数据执行每个交易业务, 得到每个交易业务的交易执行结果。

可选的, 如图 6 所示, 若区块链节点在本地缓存中未查找到与交易业务相关联的目标数据, 则区块链节点还可以通过业务合约, 从图 6 所示的分布式数据库中查找与该交易业务相关联的目标数据, 进而可以将从分布式数据库中查找到的目标数据作为该交易业务的目标读取数据。

在本申请实施例中, 本地缓存对应的本地数据库的数据读取等级优于图 6 所示的分布式数据库的数据读取等级。其中, 如图 6 所示的分布式数据库中也可以包括主数据库和备数据库, 通过在分布式数据库中引入一主多备模式, 可以使用更多的存储设备来支撑海量数据的业务场景, 进而可以在提升数据存储量的同时, 极大地增强分布式数据库的可用性。

区块链节点可以在节点本地的存储层中创建一个本地缓存, 这样, 该创建的本地缓存可以用于缓存由执行层发送来的已经执行完成的区块的块数据 (例如, 已执行完的区块的块数据可以为已执行完的区块 A 的块数据)。

进一步的, 当该区块链节点的共识层提交下一个区块 (例如, 区块 A 的下一区块为区块 B) 时, 可以将该待执行的区块 B 统称为上述第一区块, 进而可以在执行层中执行该第一区块中的 K1 个交易业务, 比如, 该执行层执行该第一区块时, 可以优先从前述本地缓存中缓存的已执行完成的区块的块数据, 读取用于执行对应交易业务的目标数据, 进而可以将本地缓存中所读取到的目标数据作为对应交易业务的目标读取数据, 进而可以根据对应交易业务的目标读取数据执行对应交易业务, 以得到相应交易业务的交易执行结果。可选的, 如果该区块链节点的执行层在执行某个交易业务时, 没有在本本地缓存中读取到 (即命中) 用于在当前时刻执行交易业务的目标数据, 则可以进一步通过网络从分布式数据库中查找与这个交易业务相关联的目标数据, 进而可以将分布式数据库中所查找到的目标数据作为这个交易业务的目标读取数据。

然后, 该区块链节点的存储层可以将上述执行层发送的第一区块的区块头信息 (即第一区块头信息)、该第一区块的交易列表以及该第一区块中的每个交易业务的交易执行结果, 统称为该第一区块的第一块数据, 进而将第一

区块的第一块数据作为上述执行层当前已执行完成的区块的块数据。然后，该区块链节点的存储层可以将当前已执行完成的区块的块数据（即第一区块的第一块数据）进一步写入至本地缓存和本地存储。

为便于理解，进一步的，请参见图 7，图 7 是本申请实施例提供的一种存储层的场景示意图。如图 7 所示的存储层 700 可以为上述图 2 对应的实施例中的存储层 300。在使用分布式数据库替换现有技术中的本地存储的实现方式中，为了解决分布式存储存在读、写效率较差的问题，本申请实施例提出了在图 7 所示的存储层 700 中扩展创建多个模块。这里的多个模块具体可以包括图 7 所示的灾备模块 710、本地缓存模块 720 和持久化模块 730。其中，灾备模块 710 可以与上述图 2 对应的实施例中的灾备模块 310 具有相同功能；本地缓存模块 720 可以与上述图 2 对应的实施例中的本地缓存模块 320 具有相同功能；持久化模块 730 可以与上述图 2 对应的实施例中的持久化模块 330 具有相同功能。

其中，该灾备模块 710 中具体可以包括图 7 所示的数据打包单元 711、本地存储单元 712、块存储状态映射表单元 713，版本刷新线程 714、数据恢复线程 716 和持久化完毕块高单元 715。

灾备模块 710 的灾备处理流程可以描述如下：

数据打包单元 711 可以将当前区块（即上述第一区块）中的所有待存储的数据（比如，n1 个数据）序列化成 1 个数据（例如，当前序列化得到的数据可以为数据 V1），进而以当前区块（即第一区块）的区块高度（例如，H1）为 KV 键值对中的 key（即键），并以该数据打包单元 731 对当前区块的处理结果（即序列化得到的数据 V1）为 Value（即值），进而可以将该 KV 键值对（即（当前区块的区块高度，当前序列化得到的数据））存入图 7 所示的本地存储单元 712。比如，（当前区块的区块高度，当前序列化得到的数据）=（H1，V1）。应当理解，本申请实施例可以将当前区块（即上述第一区块）中的所有待存储的 n1 个数据统称为区块高度为 H1 的第一区块的待存数据量。比如，这 n1 个数据可以包括但不限于上述 K1 个交易业务的交易执行结果。

块存储状态映射表单元 713 可以用于存储对应区块的待存数据量，当对应区块的待存数据量为 0 时，可以用于表示该区块链节点当前已经完成对这个区块的分布式存储。比如，当存储处理池（例如，图 7 所示的存储处理池 732）在上述重叠检测后保留的数据中存在成功写入上述分布式数据库的数据所属的块数据（即上述目标块数据）时，可以在该块存储状态映射表单元 713 所存储的存储状态映射表中，将该目标块数据对应的存储数据量进行递减处理，以实时对该存储状态映射表中待存数据量进行更新处理。

持久化完毕块高单元 715 可以用于存放当前分布式数据库中连续、且完整的区块的块数据。比如，若分布式数据库中已经存完的区块包括：区块 0-区块 100，区块 111、区块 112、区块 113。则该持久化完毕块高单元 715 可

以用于存储区块 100（此时，区块 100 的块高可以为 H），因为从区块 100 至区块 111 的数据并不连续，因此，该持久化完毕块高单元 715 可以定时查询图 7 所示的块存储状态映射表单元 713 维护的块存储状态映射表中的块高 H+1 是否存完，如果存完，则该持久化完毕块高单元 715 可以将其所维护的持久化完毕块高更新为 $H=H+1$ （即此时，新的 H 可以为 $100+1=101$ ）。与此同时，区块链节点会调用图 7 所示的版本刷新线程 714 将块高 H（例如，前述存放的区块 100 的块高）存入至分布式数据库。

这里的版本刷新线程 714 可以将图 7 所示的持久化完毕块高单元 715 中的块高 H 存入至上述分布式数据库，之所以要存入分布式数据库是因为本地节点有可能出现损坏现象，比如，系统故障、硬盘损坏等。所以，当该区块链节点在后续进行使用时，可以从其他共识节点处获取相应的区块高度所对应区块中的数据进行补全。

数据恢复线程 716 可以在节点重启时，对图 7 所示的本地存储单元 712 中的最新块高与分布式数据库中的最新块高（即由前述版本刷新线程所存入的块高）进行比对，如果发现远程的分布式数据库有区块欠缺现象（即区块高度不一致，比如，本地存储单元 712 的最新块高高于分布式数据库中的最新块高），则该区块链节点可以从图 7 所示的本地存储单元中获取欠缺的块高数据，并可以按照块高的由高到底的顺序将欠缺的块高数据重新写入到分布式数据库。这样，当数据恢复完毕后，则可以删除图 7 所示的本地存储单元 712 中的对应数据，以防止本地存储单元 712 对应的本地存储空间的过度消耗。

本地存储单元 712 与上述图 2 所示实施例中的本地存储 312 具有相同功能，比如，该本地存储单元 712 可以用于存储完整的灾备数据（即存储完整的备用数据），具体的，该本地存储单元 712 中的灾备数据可以为上述执行层发送的执行完毕的当前区块的块数据。

为便于理解，进一步的，请参见图 8，图 8 是本申请实施例提供的一种数据打包单元的场景示意图。如图 8 所示的数据打包单元 811 可以为前述图 7 所示实施例中的用于进行数据打包的数据打包单元 711。

其中，当执行层执行的当前区块为图 8 所示的块高 H1 对应的区块 1 时，可以将该区块 1 称之为第一区块。此时，如图 8 所示，当执行层执行完区块 1 后，该数据打包单元 811 可以将当前区块（即上述第一区块）中的所有待存储的数据（比如，图 8 所示的区块头 11、交易集 12 和块写数据集 13）序列化成一个数据（例如，当前序列化得到的数据可以为图 8 所示的区块数据集 1），进而可以将当前区块（即第一区块）的区块高度（例如，图 8 所示的块高 H1）作为 KV 键值对中的 key（即键），并可以将数据打包单元 831 对当前区块的处理结果（即序列化得到的区块数据集 1）作为 Value（即值），进而可以将该 KV 键值对（即（当前区块的区块高度，当前序列化得到的数据））

写入图 8 所示的本地存储单元 812。比如，此时，（当前区块的区块高度，当前序列化得到的数据）=（块高 H1，区块数据集 1）。应当理解，当前区块（即上述第一区块）中的所有待存储的 n1 个数据可以具体可以包括图 8 所示的区块头 11、交易集 12 和块写数据集 13。

- 5 以此类推，如图 8 所示，当执行层执行的当前区块为图 8 所示的块高 H2 所对应的区块 2 时，可以将区块 2 称之为新的第一区块，该新的第一区块可以为第二区块。此时，如图 8 所示，当执行层执行完图 8 所示的区块 2 时，该数据打包单元 811 可以将当前区块（即新的第一区块）中的所有待存储的数据（比如，图 8 所示的区块头 21、交易集 22 和块写数据集 23）序列化成 10 1 个数据（例如，当前序列化得到的数据可以为图 8 所示的区块数据集 2），进而可以将当前区块（即新的第一区块）的区块高度（例如，图 8 所示的块高 H2）作为 KV 键值对中的 key（即键），并可以将数据打包单元 831 对当前区块的处理结果（即序列化得到的区块数据集 2）作为 Value（即值），进而可以将该 KV 键值对（即（当前区块的区块高度，当前序列化得到的数据））
15 写入图 8 所示的本地存储单元 812。比如，此时，（当前区块的区块高度，当前序列化得到的数据）=（块高 H2，区块数据集 2）。应当理解，当前区块（即新的第一区块）中的所有待存储的 n1 个数据可以具体可以包括图 8 所示的区块头 21、交易集 22 和块写数据集 23。

- 由此可见，该数据打包单元 811 可以负责将每个区块的所有数据打包成一个区块数据集，进而可以将块高（即区块高度）作为 key，将区块数据集作为 value，存入至图 8 所示的本地存储单元 812。这样，当系统宕机重启后，该区块链节点可以对比图 8 所示的本地存储单元 812 存储的最新块高与分布式数据库存储的最新块高，以将远程分布式数据库所欠缺的相应块高所对应区块的数据进行数据补全。其中，可以理解的是，该区块链节点可以通过
25 getLastBlockHeight 接口（即一种通用区块链接口）从上述分布式数据库中获取到分布式数据库的最新块高。

- 该区块链节点之所以需要将该第一区块的第一块数据写入至本地存储，是因为这里的本地缓存为该区块链节点的系统内存中的存储单元，而该本地存储可以为该区块链节点的存储硬盘中的存储单元。当区块链节点因为系统
30 故障而遭受宕机或者重启时，不会因为本地缓存中存储的内容的消失，而没有办法直接基于重启后的本地缓存来读取到目标数据。所以，通过引入该灾备模块可以在该区块链节点出现故障时，智能实现节点内的主备切换，进而在读取目标数据的过程中，可以通过备用的本地存储提升容灾能力，以便后续在执行完第一区块之后，可以快速且准确地将当前执行完的区块的块数据
35 写入到分布式数据库。

 步骤 S203，将第一区块头信息、交易列表以及每个交易业务的交易执行结果，作为第一区块的第一块数据。

步骤 S204, 将第一区块的第一块数据写入本地缓存, 以及将第一区块的第一块数据写入本地缓存对应的本地存储。

5 区块链节点执行完步骤 S204 后, 可以执行下述步骤 S205, 此外, 该区块链节点还可以在将第一块数据写入本地缓存和本地存储后, 进一步通知前述执行层执行第二区块中的 K2 个交易业务; 其中, 第二区块可以是由共识层对 K2 个交易业务进行打包得到的; 这里的 K2 可以为正整数; 可以理解的是, 这里的第二区块为前述第一区块的下一区块。

10 由此可见, 本申请实施例中的区块链节点可以支持读写分离, 这样, 可以减轻对数据库 (这里指分布式数据库) 的访问压力。即区块链节点在执行完第一区块之后, 一方面可以将该第一区块的第一块数据写入分布式数据库, 另一方面还可以在执行第二区块时, 优先从本地缓存中查找 (即读取) 与 K2 个交易业务中的每个交易业务相关联的目标数据, 进而大大提前对下一区块的处理时间, 增加该区块链节点所维护的区块链的吞吐量。

15 其中, 步骤 S201-步骤 S204 的具体实现方式, 可以参见上述图 3 所示实施例中将对第一块数据写入至本地缓存和本地存储的具体过程的描述, 这里不再进行赘述。

步骤 S205, 将第一区块的第一块数据写入本地缓存和本地存储后, 检测与本地缓存相关联的存储缓冲池的第一区块处理空间, 得到第一空间检测结果。

20 其中, 存储缓冲池中包括 N 个待存块数据; N 为非负整数, 且 N 用于表征第一区块处理空间的待存区块数量。

25 区块链节点在将第一块数据写入存储缓冲池之前, 需要通过第一刷新线程判断存储缓冲池是否具备处理下一区块的空间, 即该区块链节点可以通过第一刷新线程判断该存储缓冲池的第一区块处理空间中的待存区块数量是否达到缓冲区块处理阈值。如果该区块链节点判断待存区块数量已达到缓冲区块处理阈值, 则该区块链节点可以通知执行层对第一块数据进行存储等待。反之, 如果该区块链节点判断出待存区块数量未达到缓冲区块处理阈值, 则该区块链节点可以进一步执行下述步骤 S206。

30 步骤 S206, 若第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值, 则基于 N 个待存块数据, 将第一块数据写入存储缓冲池。

35 具体的, 区块链节点在执行完上述步骤 S205 之后, 若确定出第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值, 则可以进一步从存储缓冲池中获取 N 个待存块数据, 进而可以在 N 个待存块数据中查找与第一块数据存在数据重叠的待存块数据; 进一步的, 若在 N 个待存块数据中查找到与第一块数据存在数据重叠的待存块数据, 则区块链节点可以将查找到的待存块数据作为重合块数据; 进

一步的，区块链节点可以在N个待存块数据中将重合块数据替换为第一块数据，将替换后的N个待存块数据作为与第一块数据相关联的第一合并数据，将第一合并数据写入存储缓冲池。区块链节点若在N个待存块数据中未查找到与第一块数据存在数据重叠的待存块数据，则将第一块数据和N个待存块数据作为与第一块数据相关联的第二合并数据，将第二合并数据写入存储缓冲池，且在存储缓冲池中对第一区块处理空间的待存区块数量进行递增处理；递增处理后的第一区块处理空间的待存块数据的数量为N+1。

由此可见，在本申请实施例中，区块链节点在将第一块数据写入存储缓冲池的过程中，需要进行合并处理，即需要判断该第一块数据是否与存储缓冲池中的N个待存块数据存在数据重叠。如果存在数据重叠，则需要进行数据合并，即可以用第一块数据替换掉N个待存块数据中存在数据重叠的待存块数据，进而可以将合并得到的第一合并数据写入存储缓冲池，从而可以避免因相同的数据（即存在数据重叠的两个数据）重复存储至后续分布式数据库而造成存储空间浪费的现象。如果不存在数据重叠，则可以将第一块数据与N个待存块数据进行合并处理，以将合并得到的第二合并数据写入到存储缓冲池。应当理解，本申请实施例通过使用合并处理的方式，合并区块重复数据，可以减少数据写入量。

可选的，区块链节点在执行完上述步骤S206时，即该区块链节点在将第一块数据写入存储缓冲池时，还可以接收由执行层发送的第二区块的第二块数据；这里的第二块数据是上述执行层在执行完K2个交易业务之后得到的；进一步的，区块链节点可以通过第一刷新线程对存储缓冲池的第一区块处理空间进行缓冲刷新，得到缓冲刷新结果，且用缓冲刷新结果更新第一空间检测结果；进一步的，若更新后的第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到存储缓冲池的缓冲区块处理阈值，则区块链节点将第二块数据添加至第一块数据所在的存储缓冲池。

应当理解，在本申请实施例中，通过在存储层的持久化模块中引入存储缓冲池，可以尽可能地防止区块执行过快。即该存储层中的存储缓冲池可以用于将数据成功存入分布式数据库之后，通知本地缓存淘汰某些在历史缓存时间比较久的数据，这样就可以从根源上解决本地缓存因区块执行过快而淘汰掉尚未存储分布式数据库的数据，从而避免造成执行层读取到的数据存在错误的现象。

步骤S207，检测与存储缓冲池相关联的存储处理池的第二区块处理空间，得到第二空间检测结果。

其中，存储处理池用于存储与分布式数据库相关联的M个待反馈块数据；M为非负整数，且M用于表征第二区块处理空间的待反馈区块数量。

在本申请实施例中，区块链节点在执行上述步骤S205-步骤S206的过程中，还可以异步执行步骤S207-步骤S208。比如，该区块链节点可以在将第

一块数据写入至存储缓冲池的过程中，异步对该存储处理池的第二区块处理空间进行空间检测，以得到第二空间检测结果。

5 若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量达到第二区块处理空间的反馈区块处理阈值，则确定该存储处理池当前不具备保持数据的空间，此时，该区块链节点为了防止一次性写入过多数据至存储处理池，则需要通知存储缓冲池对待存数据进行存储等待，即此时，该区块链节点无需指示存储处理池从存储缓冲池中获取待合并块数据。

10 反之，若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值，则确定该存储处理池当前具备保持数据的空间，此时，该区块链节点可以执行下述步骤 S208，以从存储缓冲池中获取待合并块数据。

15 步骤 S208，若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值，则基于待反馈块数据的数量和反馈区块处理阈值，从存储缓冲池中确定待合并块数据，对待合并块数据和 M 个待反馈块数据进行重叠检测，将重叠检测后保留的数据写入分布式数据库。

20 具体的，在存储缓冲池中的待存数据包括 N 个待存块数据和第一块数据，且第一块数据与 N 个待存块数据之间不存在数据重叠时，区块链节点若判断出第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值，则可以确定第二区块处理空间中待反馈块数据的数量与反馈区块处理阈值之间的数量差值，进而可以基于数量差值以及存储缓冲池中的待存数据的区块高度，从 N 个待存块数据和第一块数据中确定待合并块数据，并可以对待合并块数据和 M 个待反馈块数据进行重叠检测，得到重叠检测结果；此时，若区块链节点判断出重叠检测结果指示
25 M 个待反馈块数据中不存在与待合并块数据相重叠的待反馈块数据，则可以将待合并块数据写入存储处理池，进而可以将存储处理池中的待合并块数据和 M 个待反馈块数据作为重叠检测后保留的数据，以便于后续可以将重叠检测后保留的数据写入分布式数据库。若区块链节点判断出重叠检测结果指示
30 M 个待反馈块数据中存在与待合并块数据相重叠的待反馈块数据，则需要用从该存储缓冲池获取到的待合并数据覆盖与其存在重叠的待反馈块数据，以防止重复数据的多次写入。

步骤 S209，接收分布式数据库反馈的目标块数据的数据完成量。

其中，目标块数据是重叠检测后保留的数据中成功写入分布式数据库的数据所属的块数据。

35 步骤 S210，基于数据完成量，在块存储状态映射表中对目标块数据对应的待存数据量进行递减处理，直到待存数据量达到递减阈值为止，确定已成功将目标块数据对应的完整区块写入分布式数据库。

其中，步骤 S205-步骤 S210 的具体实现方式，可以参见上述图 3 所示实施例中多个块数据并行存入分布式数据库的具体过程的描述，这里不再赘述。

5 为便于理解，进一步的，请参见图 9，图 9 是本申请实施例提供的一种数据交互图。其中，图 9 所示的执行层 200 可以为上述图 2 所示实施例中的执行层 200。其中，区块链节点在执行图 9 所示步骤 S11 时，该区块链节点中的存储层可以接收执行层 200 在执行完第一区块时发送的第一块数据，进而可以将接收到的第一块数据保存至该存储层中的本地存储和本地缓存。区块链节点将第一块数据保存至本地存储和本地缓存后，可以进一步执行图 9 10 所示的步骤 S12，以判断该存储缓冲池 331 是否有处理当前块的块数据（即第一块数据）的空间，如果有，则可以进一步执行图 9 所示的步骤 S13。反之，则可以跳转执行图 9 所示的步骤 S11，即此时，该区块链节点需要等待该存储缓冲池具有处理第一块数据的空间时，再将该第一块数据写入到图 9 所示的存储缓冲池 331。进一步的，如图 9 所示，该区块链节点在执行步骤 15 S12 和步骤 S13 的过程中，可以异步执行图 9 所示的步骤 S14 和步骤 S15。即该区块链节点可以异步检测图 9 所示的存储处理池 332 是否具有保存数据的空间，如果有，则可以将从存储缓冲池 331 获取的数据（即上述待合并块数据）和该存储处理池当前存储的 M 个待反馈块数据进行重叠检测，以得到上述重叠检测结果。若区块链节点基于该重叠检测结果确定 M 个待反馈块数据中不存在与待合并块数据相重叠的待反馈块数据，则可以将待合并块数据 20 写入存储处理池，进而可以将存储处理池中的待合并块数据和 M 个待反馈块数据一并作为重叠检测后保留的数据，以执行图 9 所示的步骤 S16。反之，该区块链节点可以执行图 9 所示的步骤 S17，比如，区块链节点在基于重叠检测结果确定 M 个待反馈块数据中存在与待合并块数据相重叠的待反馈块数据，则可以用从该存储缓冲池 331 中获取到的数据（即新数据）覆盖更存储 25 处理池 332 中的数据（即旧数据），进而可以限制数据写入的次数，进而可以确保后续区块的交易执行结果的准确性。

由此可见，本申请实施例通过使用分布式存储，可以在存储处理池中实现并行存储，这样，可以有效地突破区块存储时的块链结构，进而在提升数据 30 存储量的同时，也在很大程度上提升数据的存储效率。另外，在执行层执行区块中的交易业务的过程中，通过使用本地缓存可以有效地减轻对分布式数据库的访问压力，进而可以在一定情况下减轻使用分布式数据库带来的网络开销。再者，通过在存储层中进行主备切换，可以在节点出现故障时有效地提升该区块链节点的容灾能力。

35 进一步的，请参见图 10，图 10 是本申请提供的一种基于分布式存储的数据处理装置的结构示意图。如图 10 所示，基于分布式存储的数据处理装置 1 可应用于上述共识网络中任意一个区块链节点，例如，基于分布式存储的

数据处理装置 1 可应用于上述图 1 所示实施例中的节点 10c。应当理解，该基于分布式存储的数据处理装置 1 可以是运行于区块链节点（比如，前述节点 10c）中的一个计算机程序（包括程序代码），例如该基于分布式存储的数据处理装置 1 可以为一个应用软件；可以理解的是，该基于分布式存储的数据处理装置 1 可以用于执行本申请实施例提供的方法中的相应步骤。如图 10 所示，基于分布式存储的数据处理装置 1 可以包括：第一检测模块 11，第一写入模块 12，第二检测模块 13 和第二写入模块 14；可选的，该基于分布式存储的数据处理装置 1 还可以包括：交易获取模块 15，交易执行模块 16，第一块确定模块 17，本地存储模块 18，通知区块执行模块 19，第二块确定模块 20，缓冲刷新模块 21，块数据添加模块 22，执行等待模块 23，待存数据确定模块 24，缓冲等待模块 25，完成量接收模块 26 和数据量递减模块 27。

第一检测模块 11，用于将第一区块的第一块数据写入本地缓存和本地存储后，检测与本地缓存相关联的存储缓冲池的第一区块处理空间，得到第一空间检测结果；存储缓冲池中包括 N 个待存块数据，N 为非负整数；

第一写入模块 12，用于若第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值，则基于 N 个待存块数据，将第一块数据写入存储缓冲池；

其中，第一写入模块 12 包括：待存块获取单元 121，第一查找确定单元 122，重合替换单元 123 和第二查找确定单元 124；

待存块获取单元 121，用于若第一空间检测结果指示第一区块处理空间中待存块数据的数量未达到第一区块处理空间的缓冲区块处理阈值，则从存储缓冲池中获取 N 个待存块数据，在 N 个待存块数据中查找与第一块数据存在数据重叠的待存块数据；

第一查找确定单元 122，用于若在 N 个待存块数据中查找到与第一块数据存在数据重叠的待存块数据，则将查找到的待存块数据作为重合块数据；

重合替换单元 123，用于将重合块数据替换为第一块数据，将替换后的 N 个待存块数据作为与第一块数据相关联的第一合并数据，将第一合并数据写入存储缓冲池。

可选的，第二查找确定单元 124，用于若在 N 个待存块数据中未查找到与第一块数据存在数据重叠的待存块数据，则将第一块数据和 N 个待存块数据作为与第一块数据相关联的第二合并数据，将第二合并数据写入存储缓冲池，且对第一区块处理空间中待存块数据的数量进行递增处理。

其中，待存块获取单元 121，第一查找确定单元 122，重合替换单元 123 和第二查找确定单元 124 的具体实现方式，可以参见上述图 3 所示实施例中对步骤 S102 的描述，这里不再进行赘述。

第二检测模块 13，用于检测与存储缓冲池相关联的存储处理池的第二区块处理空间进行空间检测，得到第二空间检测结果；存储处理池用于存储与

分布式数据库相关联的 M 个待反馈块数据, M 为非负整数;

第二写入模块 14, 用于若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量未达到第二区块处理空间的反馈区块处理阈值, 则基于待反馈块数据的数量和反馈区块处理阈值, 在存储缓冲池中确定待合并块数据, 对待合并块数据和 M 个待反馈块数据进行重叠检测, 将重叠检测后保留的数据写入分布式数据库。

其中, 存储缓冲池中的待存数据包括 N 个待存块数据和第一块数据, 且第一块数据与 N 个待存块数据之间不存在数据重叠;

第二写入模块 14 包括: 差值确定单元 141, 重叠检测单元 142, 第一写入单元 143 和第二写入单元 144;

差值确定单元 141, 用于若第二空间检测结果指示待反馈区块数量未达到第二区块处理空间的反馈区块处理阈值, 则确定待反馈块数据的数量与反馈区块处理阈值之间的数量差值;

重叠检测单元 142, 用于基于数量差值以及存储缓冲池中的待存数据的区块高度, 从 N 个待存块数据和第一块数据中确定待合并块数据, 对待合并块数据和 M 个待反馈块数据进行重叠检测, 得到重叠检测结果;

第一写入单元 143, 用于若重叠检测结果指示 M 个待反馈块数据中不存在与待合并块数据相重叠的待反馈块数据, 则将待合并块数据写入存储处理池;

第二写入单元 144, 用于将存储处理池中的待合并块数据和 M 个待反馈块数据作为重叠检测后保留的数据, 将重叠检测后保留的数据写入分布式数据库。

其中, 差值确定单元 141, 重叠检测单元 142, 第一写入单元 143 和第二写入单元 144 的具体实现方式, 可以参见上述图 3 所示实施例对步骤 S104 的描述, 这里不再进行赘述。

可选的, 交易获取模块 15, 用于在执行层接收到共识层发送的第一区块后, 获取第一区块中的 $K1$ 个交易业务; 第一区块包括第一区块头信息、与 $K1$ 个交易业务相关联的交易列表; $K1$ 为正整数;

交易执行模块 16, 用于针对 $K1$ 个交易业务中的每个交易业务, 获取用于执行该交易业务的业务合约, 通过该交易业务的业务合约执行该交易业务, 得到该交易业务的交易执行结果;

其中, 交易执行模块 16 包括: 合约获取单元 161, 合约调用单元 162, 本地查找单元 163、交易执行单元 164 和分布式查找单元 165;

合约获取单元 161, 用于基于该交易业务的合约调用地址, 获取用于执行该交易业务的业务合约;

合约调用单元 162, 用于通过该交易业务的业务合约, 从本地缓存中查找与该交易业务相关联的目标数据;

本地查找单元 163, 用于若在本本地缓存中查找到与该交易业务相关联的目标数据, 则将在本地缓存中查找到的目标数据作为该交易业务的目标读取数据;

5 交易执行单元 164, 用于基于该交易业务的目标读取数据执行该交易业务, 得到该交易业务的交易执行结果。

可选的, 分布式查找单元 165, 用于若在本本地缓存中未查找到与该交易业务的相关联的目标数据, 则通过业务合约从分布式数据库中查找与该交易业务相关联的目标数据, 将从分布式数据库中查找到的目标数据作为该交易业务的目标读取数据。

10 其中, 合约获取单元 161, 合约调用单元 162, 本地查找单元 163、交易执行单元 164 和分布式查找单元 165 具体实现方式, 可以参见上述图 3 所示实施例中对执行每个交易业务的具体过程的描述, 这里不再进行赘述。

第一块确定模块 17, 用于将第一区块头信息、交易列表、以及 K1 个交易业务的交易执行结果, 作为第一区块的第一块数据;

15 本地存储模块 18, 用于将第一区块的第一块数据写入本地缓存, 以及将第一区块的第一块数据写入本地缓存对应的本地存储。

可选的, 通知区块执行模块 19, 用于在将第一块数据写入本地缓存和本地存储后, 通知执行层执行第二区块中的 K2 个交易业务; 第二区块为第一区块的下一区块, 第二区块是由共识层对 K2 个交易业务进行打包得到的,

20 K2 为正整数。

第二块确定模块 20, 用于将第一块数据写入存储缓冲池后, 接收执行层发送的第二区块的第二块数据; 第二块数据是执行层在执行完 K2 个交易业务后得到的;

25 缓冲刷新模块 21, 用于通过第一刷新线程, 对存储缓冲池的第一区块处理空间进行缓冲刷新, 得到缓冲刷新结果, 利用缓冲刷新结果更新第一空间检测结果;

块数据添加模块 22, 用于若更新后的第一空间检测结果指示述第一区块处理空间中待存块数据的数量未达到缓冲区块处理阈值, 则基于所述第一区块处理空间中的待存块数据, 将第二块数据写入存储缓冲池。

30 可选的, 其中, 在将第一块数据写入存储缓冲池之前, 执行等待模块 23, 用于若第一空间检测结果指示第一区块处理空间中待存块数据的数量已达到缓冲区块处理阈值, 则通知执行层对第一块数据进行存储等待。

可选的, 待存数据确定模块 24, 用于将存储缓冲池中的合并数据作为待存数据;

35 缓冲等待模块 25, 用于若第二空间检测结果指示第二区块处理空间中待反馈块数据的数量已达到反馈区块处理阈值, 则通知存储缓冲池对待存数据进行存储等待。

可选的，完成量接收模块 26，用于接收分布式数据库反馈的目标块数据的数据完成量；目标块数据是重叠检测后保留的数据中成功写入分布式数据库的数据所属的块数据；

5 数据量递减模块 27，用于基于数据完成量，在块存储状态映射表中对目标块数据对应的待存数据量进行递减处理，直到待存数据量达到递减阈值为止，确定已成功将目标块数据对应的完整区块写入到分布式数据库。

其中，第一检测模块 11，第一写入模块 12，第二检测模块 13 和第二写入模块 14 的具体实现方式，可以参见上述图 3 所示实施例中对步骤 S101-步骤 S104 的描述，这里不再进行赘述。进一步的，交易获取模块 15，交易执行模块 16，第一块确定模块 17，本地存储模块 18，通知区块执行模块 19，第二块确定模块 20，缓冲刷新模块 21，块数据添加模块 22，执行等待模块 23，待存数据确定模块 24，缓冲等待模块 25，完成量接收模块 26 和数据量递减模块 27 的具体实现方式，可以参见上述图 5 所示实施例对步骤 S201-步骤 S208 的描述，这里不再进行赘述。可以理解的是，对采用相同方法的有益效果描述，也不再进行赘述。

进一步地，请参见图 11，图 11 是本申请实施例提供的一种计算机设备的结构示意图。如图 11 所示，计算机设备 1000 可以应用于上述图 1 对应实施例中的区块链节点，计算机设备 1000 可以包括：处理器 1001，网络接口 1004 和存储器 1005，此外，计算机设备 1000 还可以包括：用户接口 1003，和至少一个通信总线 1002。其中，通信总线 1002 用于实现这些组件之间的连接通信。其中，用户接口 1003 可以包括显示屏 (Display)、键盘 (Keyboard)，可选用户接口 1003 还可以包括标准的有线接口、无线接口。网络接口 1004 可选的可以包括标准的有线接口、无线接口 (如 WI-FI 接口)。存储器 1004 可以是高速 RAM 存储器，也可以是非不稳定的存储器 (non-volatile memory)，例如至少一个磁盘存储器。存储器 1005 可选的还可以是至少一个位于远离前述处理器 1001 的存储装置。如图 11 所示，作为一种计算机存储介质的存储器 1005 中可以包括操作系统、网络通信模块、用户接口模块以及设备控制应用程序。

该计算机设备 1000 中的网络接口 1004 还可以提供网络通讯功能，且可选用户接口 1003 还可以包括显示屏 (Display)、键盘 (Keyboard)。在图 11 所示的计算机设备 1000 中，网络接口 1004 可提供网络通讯功能；而用户接口 1003 主要用于为用户提供输入的接口；而处理器 1001 可以用于调用存储器 1005 中存储的设备控制应用程序，以实现前文图 3 或者图 5 所示实施例中对基于分布式存储的数据处理方法的描述，也可执行前文图 10 所示实施例中对基于分布式存储的数据处理装置 1 的描述，在此不再赘述。另外，对采用相同方法的有益效果描述，也不再进行赘述。

此外，这里需要指出的是：本申请实施例还提供了一种计算机可读存储

介质，且计算机可读存储介质中存储有前文提及的基于分布式存储的数据处理装置 1 所执行的计算机程序，且计算机程序包括程序指令，当处理器执行程序指令时，能够执行前文图 3 或者图 5 所对应实施例中对基于分布式存储的数据处理方法的描述，因此，这里将不再进行赘述。另外，对采用相同方法的有益效果描述，也不再进行赘述。对于本申请所涉及的计算机可读存储介质实施例中未披露的技术细节，请参照本申请方法实施例的描述。作为示例，程序指令可被部署在一个计算设备上执行，或者在位于一个地点的多个计算设备上执行，又或者，在分布在多个地点且通过通信网络互连的多个计算设备上执行，分布在多个地点且通过通信网络互连的多个计算设备可以组成区块链系统。

此外，需要说明的是：本申请实施例还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者计算机程序可以包括计算机指令，该计算机指令可以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，处理器可以执行该计算机指令，使得该计算机设备执行前文图 3 或图 5 所对应实施例中对基于分布式存储的数据处理方法的描述，因此，这里将不再进行赘述。另外，对采用相同方法的有益效果描述，也不再进行赘述。对于本申请所涉及的计算机程序产品或者计算机程序实施例中未披露的技术细节，请参照本申请方法实施例的描述。

本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机程序来指令相关的硬件来完成，计算机程序可存储于一计算机可读取存储介质中，该程序在执行时，可包括如上述各方法的实施例的流程。其中，存储介质可为磁碟、光盘、只读存储器（Read-Only Memory, ROM）或随机存储器（Random Access Memory, RAM）等。

以上所揭露的仅为本申请较佳实施例而已，当然不能以此来限定本申请之权利范围，因此依本申请权利要求所作的等同变化，仍属本申请所涵盖的范围。

权 利 要 求

1、一种基于分布式存储的数据处理方法，由区块链节点执行，包括：

5 将第一区块的第一块数据写入本地缓存和本地存储后，检测与所述本地缓存相关联的存储缓冲池的第一区块处理空间，得到第一空间检测结果；所述存储缓冲池中包括 N 个待存块数据，所述 N 为非负整数；

若所述第一空间检测结果指示所述第一区块处理空间中待存块数据的数量未达到所述第一区块处理空间的缓冲区块处理阈值，则基于所述 N 个待存块数据，将所述第一块数据写入所述存储缓冲池；

10 检测与所述存储缓冲池相关联的存储处理池的第二区块处理空间，得到第二空间检测结果；所述存储处理池用于存储与分布式数据库相关联的 M 个待反馈块数据，所述 M 为非负整数；

若所述第二空间检测结果指示所述第二区块处理空间中待反馈块数据的数量未达到所述第二区块处理空间的反馈区块处理阈值，则基于所述待反馈块数据的数量和所述反馈区块处理阈值，在所述存储缓冲池中确定待合并块数据，对所述待合并块数据和所述 M 个待反馈块数据进行重叠检测，将重叠检测后保留的数据写入所述分布式数据库。

2、根据权利要求 1 所述的方法，所述方法还包括：

20 执行层接收到共识层发送的第一区块后，获取所述第一区块中的 K1 个交易业务；所述第一区块包括第一区块头信息、与所述 K1 个交易业务相关联的交易列表，所述 K1 为正整数；

针对所述 K1 个交易业务中的每个交易业务，获取用于执行所述交易业务的业务合约，通过所述交易业务的业务合约执行所述交易业务，得到所述交易业务的交易执行结果；

25 将所述第一区块头信息、所述交易列表、以及所述 K1 个交易业务各自的交易执行结果，作为所述第一区块的第一块数据；

将第一区块的第一块数据写入本地缓存，以及将所述第一区块的第一块数据写入所述本地缓存对应的本地存储。

30 3、根据权利要求 2 所述的方法，所述获取用于执行所述交易业务的业务合约，通过所述交易业务的业务合约执行所述交易业务，得到所述交易业务的交易执行结果，包括：

基于所述交易业务的合约调用地址，获取用于执行所述交易业务的业务合约；

通过所述交易业务的业务合约，从所述本地缓存中查找与所述交易业务相关联的目标数据；

35 若所述本地缓存中查找到与所述交易业务相关联的目标数据，则将在所述目标数据作为所述交易业务的目标读取数据；

基于所述交易业务的目标读取数据执行所述交易业务，得到所述交易业

务的交易执行结果。

4、根据权利要求3所述的方法，所述方法还包括：

若在所述本地缓存中未查找到与所述交易业务相关联的目标数据，则通过所述业务合约，从所述分布式数据库中查找与所述交易业务相关联的目标数据，将所述目标数据作为所述交易业务的目标读取数据。

5 5、根据权利要求2所述的方法，所述方法还包括：

将所述第一块数据写入所述本地缓存和所述本地存储后，通知所述执行层执行第二区块中的K2个交易业务；所述第二区块为所述第一区块的下一区块，所述第二区块是由所述共识层对所述K2个交易业务进行打包得到的，所述K2为正整数。

10 6、根据权利要求5所述的方法，所述方法还包括：

将所述第一块数据写入所述存储缓冲池后，接收所述执行层发送的所述第二区块的第二块数据；所述第二块数据是所述执行层在执行完所述K2个交易业务后得到的；

15 通过第一刷新线程，对所述存储缓冲池的第一区块处理空间进行缓冲刷新，得到缓冲刷新结果，利用所述缓冲刷新结果更新所述第一空间检测结果；

若更新后的第一空间检测结果指示所述第一区块处理空间中待存块数据的数量未达到所述缓冲区块处理阈值，则基于所述第一区块处理空间中的待存块数据，将所述第二块数据写入所述存储缓冲池。

20 7、根据权利要求1所述的方法，在将所述第一块数据写入所述存储缓冲池之前，所述方法还包括：

若所述第一空间检测结果指示所述第一区块处理空间中待存块数据的数量已达到所述缓冲区块处理阈值，则通知所述执行层对所述第一块数据进行存储等待。

25 8、根据权利要求1所述的方法，所述基于所述N个待存块数据，将所述第一块数据写入所述存储缓冲池，包括：

从所述存储缓冲池中获取所述N个待存块数据，在所述N个待存块数据中查找与所述第一块数据存在数据重叠的待存块数据；

30 若在所述N个待存块数据中查找到与所述第一块数据存在数据重叠的待存块数据，则将查找到的待存块数据作为重合块数据；

将所述重合块数据替换为所述第一块数据，将替换后的N个待存块数据作为与所述第一块数据相关联的第一合并数据，将所述第一合并数据写入所述存储缓冲池。

9、根据权利要求8所述的方法，所述方法还包括：

35 若在所述N个待存块数据中未查找到与所述第一块数据存在数据重叠的待存块数据，则将所述第一块数据和所述N个待存块数据作为与所述第一块数据相关联的第二合并数据，将所述第二合并数据写入所述存储缓冲池，且

对所述第一区块处理空间中待存块数据的数量进行递增处理。

10、根据权利要求 1 或 8 或 9 所述的方法，所述方法还包括：

将所述存储缓冲池中的合并数据作为待存数据；

5 若所述第二空间检测结果指示所述第二区块处理空间中待反馈块数据的数量已达到所述反馈区块处理阈值，则通知所述存储缓冲池对所述待存数据进行存储等待。

11、根据权利要求 1 所述的方法，所述存储缓冲池中的待存数据包括所述 N 个待存块数据和所述第一块数据，且所述第一块数据与所述 N 个待存块数据之间不存在数据重叠；

10 所述基于所述待反馈块数据的数量和所述反馈区块处理阈值，从所述存储缓冲池中确定待合并块数据，对所述待合并块数据和所述 M 个待反馈块数据进行重叠检测，将重叠检测后保留的数据写入所述分布式数据库，包括：

确定所述待反馈块数据的数量与所述反馈区块处理阈值之间的数量差值；

15 基于所述数量差值以及所述存储缓冲池中的所述待存数据的区块高度，从所述 N 个待存块数据和所述第一块数据中确定待合并块数据；对所述待合并块数据和所述 M 个待反馈块数据进行重叠检测，得到重叠检测结果；

若所述重叠检测结果指示所述 M 个待反馈块数据中不存在与所述待合并块数据相重叠的待反馈块数据，则将所述待合并块数据写入所述存储处理池；

20 将所述存储处理池中的所述待合并块数据和所述 M 个待反馈块数据作为重叠检测后保留的数据，将所述重叠检测后保留的数据写入所述分布式数据库。

12、根据权利要求 11 所述的方法，所述方法还包括：

25 接收所述分布式数据库反馈的目标块数据的数据完成量；所述目标块数据是所述重叠检测后保留的数据中成功写入所述分布式数据库的数据所属的块数据；

基于所述数据完成量，在块存储状态映射表中对所述目标块数据对应的待存数据量进行递减处理，直到所述待存数据量达到递减阈值为止，确定已成功将所述目标块数据对应的完整区块写入所述分布式数据库。

30 13、一种基于分布式存储的数据处理装置，包括：

第一检测模块，用于将第一区块的第一块数据写入本地缓存和本地存储后，检测与所述本地缓存相关联的存储缓冲池的第一区块处理空间，得到第一空间检测结果；所述存储缓冲池中包括 N 个待存块数据，所述 N 为非负整数；

35 第一写入模块，用于若所述第一空间检测结果指示所述第一区块处理空间中待存块数据的数量未达到所述第一区块处理空间的缓冲区处理阈值，则基于所述 N 个待存块数据，将所述第一块数据写入所述存储缓冲池；

第二检测模块，用于检测与所述存储缓冲池相关联的存储处理池的第二区块处理空间，得到第二空间检测结果；所述存储处理池用于存储与分布式数据库相关联的M个待反馈块数据，所述M为非负整数；

- 5 第二写入模块，用于若所述第二空间检测结果指示所述第二区块处理空间中待反馈块数据的数量未达到所述第二区块处理空间的反馈区块处理阈值，则基于所述待反馈块数据的数量和所述反馈区块处理阈值，在所述存储缓冲池中确定待合并块数据，对所述待合并块数据和所述M个待反馈块数据进行重叠检测，将重叠检测后保留的数据写入所述分布式数据库。

14、一种计算机设备，包括：处理器和存储器；

- 10 所述处理器与存储器相连，其中，所述存储器用于存储计算机程序，所述处理器用于调用所述计算机程序，以使得所述计算机设备执行权利要求1-12任一项所述的方法。

- 15 15、一种计算机可读存储介质，所述计算机可读存储介质中存储有计算机程序，该计算机程序适于由处理器加载并执行，以使得具有所述处理器的计算机设备执行权利要求1-12任一项所述的方法。

16、一种计算机程序产品，包括指令，当其在计算机上运行时，使得计算机实现如权利要求1-12中任一项所述的方法。