



(12)发明专利申请

(10)申请公布号 CN 110442644 A

(43)申请公布日 2019. 11. 12

(21)申请号 201910611644.7

(22)申请日 2019.07.08

(71)申请人 深圳壹账通智能科技有限公司  
地址 518052 广东省深圳市前海深港合作区前湾一路1号A栋201室

(72)发明人 薄辰龙

(74)专利代理机构 广州华进联合专利商标代理有限公司 44224

代理人 黄晶晶

(51) Int. Cl.

G06F 16/27(2019.01)

G06F 16/22(2019.01)

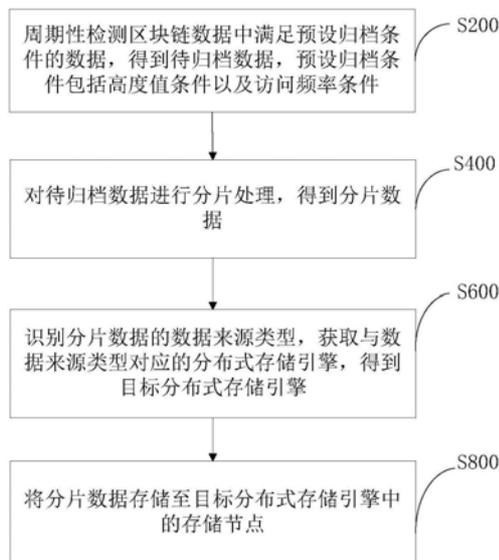
权利要求书2页 说明书11页 附图4页

(54)发明名称

区块链数据归档存储方法、装置、计算机设备和存储介质

(57)摘要

本申请涉及区块存储系统领域,提供一种区块链数据归档存储方法、装置、计算机设备和存储介质,通过周期性检测待归档数据并对其进行分片处理,完成数据归档,避免数据冗余,根据分片数据来源类型能够灵活的选择存储方式,保全数据的安全可用性,并且将分片数据分配至对应的分布式存储引擎中的存储节点,通过分布式存储的方式,实现区块链数据存储空间的扩容,满足区块链数据的线性增长对存储空间的需求,提高数据处理效率。



1. 一种区块链数据归档存储方法,其特征在于,所述方法包括:

周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,所述预设归档条件包括高度值条件以及访问频率条件;

对所述待归档数据进行分片处理,得到分片数据;

识别所述分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎;

将所述分片数据存储至所述目标分布式存储引擎中的存储节点。

2. 根据权利要求1所述的区块链数据归档存储方法,其特征在于,所述周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据包括:

按照周期时间统计区块链节点中最低高度值的数据与最高高度值的数据之间的数据总量,所述周期时间小于生成满足预设总量的数据集的所需时间;

当所述数据总量大于所述预设总量时,则从最低高度值依次获取满足预设总量的数据集;

查看所述数据集中每个数据的访问频率;

当所述数据集中每个数据均小于预设访问频率时,则将所述数据集中的数据确定为待归档数据。

3. 根据权利要求1所述的区块链数据归档存储方法,其特征在于,所述分布式存储引擎包括私有式存储引擎、联盟式存储引擎以及公有式存储引擎;所述识别所述分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎包括:

当识别到所述分片数据来源于私有区块链时,修改通用私有式存储引擎对应的入口参数,获取私有式存储引擎;

当识别到所述分片数据基于智能合约来源于联盟区块链时,修改通用联盟式存储引擎对应的入口参数,获取联盟式存储引擎;

当识别到所述分片数据来源于公有式区块链时,修改通用公有式存储引擎对应的入口参数,获取公有式存储引擎。

4. 根据权利要求1所述的区块链数据归档存储方法,其特征在于,所述将所述分片数据存储至所述目标分布式存储引擎中的存储节点包括:

根据哈希算法,计算所述分片数据的哈希值;

将所述分片数据存储至目标分布式存储引擎中所述哈希值指向的存储节点。

5. 根据权利要求1所述的区块链数据归档存储方法,其特征在于,所述将所述分片数据存储至所述目标分布式存储引擎中的存储节点包括:

对各存储节点进行评估归类,确定所述各存储节点的存储性能;

对所述分片数据进行权值分配评估,确定所述分片数据的存储需求;

根据所述各存储节点的存储性能以及所述各分片数据的存储需求,将所述分片数据存储至目标分布式存储引擎中的存储节点。

6. 根据权利要求1至5中任一项所述的区块链数据归档存储方法,其特征在于,所述分片数据所在的数据分片包括多个数据分组;所述将所述分片数据存储至所述目标分布式存储引擎中的存储节点之后,还包括:

计算所述各数据分组的哈希值;

根据所述各数据分组的哈希值,构建与各数据分组所在的数据分片对应的梅克尔树;  
记录所述各梅克尔树与所述各数据分片所在的存储节点的对应关系。

7. 根据权利要求1所述的区块链数据归档存储方法,其特征在于,还包括:

对所述各存储节点进行容量检测;

当所述各存储节点容量达到预设存储容量时,发送存储节点扩展请求。

8. 一种区块链数据归档存储装置,其特征在于,所述装置包括:

数据检测模块,用于周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,所述预设归档条件包括高度值条件以及访问频率条件;

数据分片模块,用于对所述待归档数据进行分片处理,得到分片数据;

分布式存储引擎获取模块,用于识别所述分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎;

数据存储模块,用于将所述分片数据存储至所述目标分布式存储引擎中的存储节点。

9. 一种计算机设备,包括存储器和处理器,所述存储器存储有计算机程序,其特征在于,所述处理器执行所述计算机程序时实现权利要求1至7中任一项所述方法的步骤。

10. 一种计算机可读存储介质,其上存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现权利要求1至7中任一项所述的方法的步骤。

## 区块链数据归档存储方法、装置、计算机设备和存储介质

### 技术领域

[0001] 本申请涉及数据存储技术领域,特别是涉及一种区块链数据归档存储方法、装置、计算机设备和存储介质。

### 背景技术

[0002] 随着科学技术的发展,互联网已经深入到人们的日常生活中。人们可以通过互联网查询数据、购买商品、社交等,给人们带来了巨大的便利。然而随着互联网数据的爆炸式增长,网络安全以及海量数据存储成为各个企业公司的艰巨任务,随后出现了使用区块链技术对海量数据进行记录存储的方式,保证了数据的安全性以及存储空间。

[0003] 区块链技术,也被称为分布式账本技术,是一种由若干台计算机设备共同参与记账,共同维护一份完整的分布式数据库的新技术,区块链技术具有去中心化、公开透明的特性,每台计算机设备作为区块链的节点设备可以参与数据库记录、并且各计算设备之间可以快速的进行数据同步,区块链的各节点设备通常要处理区块链系统高并发的共识、验证以及读写操作。

[0004] 但是,区块链节点存储空间随着区块增加而线性增长,区块节点存储空间有限,增量的区数据会逐渐影响存储器的响应速度,进而影响节点设备的对高并发的共识、验证以及读写操作的效率,降低了对数据处理效率。

### 发明内容

[0005] 基于此,有必要针对现有的区块链数据存储方案降低节点设备的数据处理效率的问题,提供一种区块链数据归档存储方法、装置、计算机设备和存储介质。

[0006] 一种区块链数据归档存储方法,方法包括:

[0007] 周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,预设归档条件包括高度值条件以及访问频率条件;

[0008] 对待归档数据进行分片处理,得到分片数据;

[0009] 识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎;

[0010] 将分片数据存储至目标分布式存储引擎中的存储节点。

[0011] 在其中一个实施例中,周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据包括:

[0012] 按照周期时间统计区块链节点中最低高度值的数据与最高高度值的数据之间的数据总量,周期时间小于生成满足预设总量的数据集的所需时间;

[0013] 当数据总量大于预设总量时,则从最低高度值依次获取满足预设总量的数据集;

[0014] 检测数据集中每个数据的访问频率;

[0015] 当数据集中每个数据均小于预设访问频率时,则将数据集中的数据确定为待归档数据。

[0016] 在其中一个实施例中,分布式存储引擎包括私有式存储引擎、联盟式存储引擎以及公有式存储引擎,识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎包括:

[0017] 当识别到分片数据来源于私有区块链时,修改通用私有式存储引擎对应的入口参数,获取私有式存储引擎;

[0018] 当识别到分片数据基于智能合约来源于联盟区块链时,修改通用联盟式存储引擎对应的入口参数,获取联盟式存储引擎;

[0019] 当识别到分片数据来源于公有式区块链时,修改通用公有式存储引擎对应的入口参数,获取目标公有式存储引擎。

[0020] 在其中一个实施例中,将分片数据存储至目标分布式存储引擎中的存储节点包括:

[0021] 根据哈希算法,计算分片数据的哈希值;

[0022] 将分片数据存储至目标分布式存储引擎中哈希值指向的存储节点。

[0023] 在其中一个实施例中,将分片数据存储至目标分布式存储引擎中的存储节点包括:

[0024] 对各存储节点进行评估归类,确定各存储节点的存储性能;

[0025] 对分片数据进行权值分配评估,确定分片数据的存储需求;

[0026] 根据各存储节点的存储性能以及各分片数据的存储需求,将分片数据存储至目标分布式存储引擎中的存储节点。

[0027] 在其中一个实施例中,分片数据所在的数据分片包括多个数据分组;将分片数据存储至目标分布式存储引擎中的存储节点之后,还包括:

[0028] 计算各数据分组的哈希值;

[0029] 根据各数据分组的哈希值,构建与各数据分组所在的数据分片对应的梅克尔树;

[0030] 记录各梅克尔树与各数据分片所在的存储节点的对应关系。

[0031] 在其中一个实施例中,还包括:

[0032] 对各存储节点进行容量检测;

[0033] 当各存储节点容量达到预设存储容量时,发送存储节点扩展请求。

[0034] 一种区块链数据归档存储装置,装置包括:

[0035] 数据检测模块,用于周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,预设归档条件包括高度值条件以及访问频率条件;

[0036] 数据分片模块,用于对待归档数据进行分片处理,得到分片数据;

[0037] 分布式存储引擎获取模块,用于识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎;

[0038] 数据存储模块,用于将分片数据存储至目标分布式存储引擎中的存储节点。

[0039] 一种计算机设备,包括存储器和处理器,所述存储器存储有计算机程序,所述处理器执行所述计算机程序时实现以下步骤:

[0040] 周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,预设归档条件包括高度值条件以及访问频率条件;

[0041] 对待归档数据进行分片处理,得到分片数据;

- [0042] 识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎;
- [0043] 将分片数据存储至目标分布式存储引擎中的存储节点。
- [0044] 一种计算机可读存储介质,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以下步骤:
- [0045] 周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,预设归档条件包括高度值条件以及访问频率条件;
- [0046] 对待归档数据进行分片处理,得到分片数据;
- [0047] 识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎;
- [0048] 将分片数据存储至目标分布式存储引擎中的存储节点。
- [0049] 上述区块链数据归档存储方法、装置、计算机设备和存储介质,通过周期性检测待归档数据并对其进行分片处理,完成数据归档,避免数据冗余,根据分片数据来源类型能够灵活的选择存储方式,保全数据的安全可用性,并且将分片数据分配至对应的分布式存储引擎中的存储节点,通过分布式存储的方式,实现区块链数据存储空间的扩容,满足区块链数据的线性增长对存储空间的需求,提高数据处理效率。

#### 附图说明

- [0050] 图1为一个实施例中区块链数据归档存储方法的应用环境图;
- [0051] 图2为一个实施例中区块链数据归档存储方法的流程示意图;
- [0052] 图3为另一个实施例中区块链数据归档存储方法的详细流程示意图;
- [0053] 图4为一个实施例中哈希环的示意图;
- [0054] 图5为另一个实施例中区块链数据归档存储装置的结构框图;
- [0055] 图6为另一个实施例中区块链数据归档存储装置的详细结构框图;
- [0056] 图7为一个实施例中计算机设备的内部结构图。

#### 具体实施方式

- [0057] 为了使本申请的目的、技术方案及优点更加清楚明白,以下结合附图及实施例,对本申请进行进一步详细说明。应当理解,此处描述的具体实施例仅仅用以解释本申请,并不用于限定本申请。
- [0058] 本申请提供的区块链数据归档存储方法,可以应用于如图1所示的应用环境中。区块链网络102包括多个区块链节点,区块链网络102、控制系统104以及分布式存储系统106之间通过网络实现两两通信。其中,区块链节点以及控制系统104可以是不同的计算机设备(计算机或服务器),也可以是部署在同一台计算机或服务器上的应用程序,每台计算机设备作为区块链的节点设备可以参与数据库记录、并且各计算机设备之间可以快速地进行数据同步,每一个区块链节点都有对应的控制系统104,分布式存储系统106可以包括多台存储设备,其中,多台存储设备通过应用程序或软件集合起来共同对外提供数据存储和访问功能。分布式存储系统106可以为至少一个区块链节点提供区块链块数据存储服务。其中,控制系统104以多台服务器为例进行说明,服务器周期性检测区块链数据中满足预

设归档条件的数据,得到待归档数据,对所述待归档数据进行分片处理,得到分片数据,识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎,将分片数据存储至目标分布式存储引擎中的存储节点。上述方案,在实现对区块链数据进行归档处理的同时,又通过分布式存储的方式,实现区块链数据存储空间的扩容,满足区块链数据的线性增长对存储空间的需求。其中,服务器可以用独立的服务器或者是多个服务器组成的服务器集群来实现。

[0059] 在其中一个实施例中,如图2所示,提供了一种区块链数据归档存储方法,以该方法应用于图1中的控制系统104中的服务器为例进行说明,包括以下步骤:

[0060] 步骤S200,周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,预设归档条件包括高度值条件以及访问频率条件。

[0061] 区块链,也被称为分布式账本技术,是一种由若干台计算机设备共同参与记账,共同维护一份完整的分布式数据库(可看作账本)的新技术。区块链数据是指按照时间顺序被装订成一个个区块文件(可看作帐页),并串起来形成链条形式的数据。具体的来说,区块链数据中每一个“块数据”就犹如账本中的“帐页”,每一页都记录了若干条数据交易记录,把一页页的账页按照时间顺序装订起来就形成一个完整的账本即分布式数据库。用户可通过终端区块链网络102中的区块链节点发起数据交易请求,提交待处理的数据,该区块链节点接收到数据处理请求后,对待处理的数据进行相应的数据处理,例如,与其它区块链节点进行数据的共识处理、存储等。数据归档是将不再经常使用的数据移到一个单独的存储设备来进行长期保存的过程。区块链中的块数据的高度值以及访问频率能够直观的反映出区块链节点的块数据是否是经常使用的数据(热数据)或长期未使用的数据(冷数据),故本实施中,将数据高度值条件以及访问频率条件作为检测区块链数据中待归档数据的预设归档条件。具体的,按照周期时间检测区块链数据中满足预设归档条件的数据,将满足预设归档条件的数据确定为待归档数据,预设归档条件包括高度值条件以及访问频率条件。

[0062] 步骤S400,对待归档数据进行分片处理,得到分片数据。

[0063] 当得到待归档数据之后,可以是将待归档数据划分成多个(至少两个)数据分组,将每个数据分组至少添加至两个数据分片中,保证每两个数据分片存储的数据包括所有的分组数据(待归档数据),而每一个数据分片又不包括所有的分组数据。具体的,将待归档数据进行分组,可采用平均划分的分组方式,将待归档数据分成包括相同数据量的分组数据,或者通过随机划分的方式,将待归档数据划分成包括不同数据量的分组数据,然后将划分后的每个数据分组都添加至至少两个数据分片中。当某个数据分片遭到恶意破坏时不会丢失完整数据,并且利用其它两个数据分片能对文件进行修复。

[0064] 步骤S600,识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎。

[0065] 存储引擎是用于将MySQL (Structured Query Language, 结构化查询语言) 中的数据用各种不同的技术存储在内存中,这些技术中的每一种技术都使用不同的存储机制、索引技巧、锁定水平并且最终提供广泛的不同功能和能力,通过选择不同的技术,获取额外的速度或者功能,从而改善应用整体功能。可以通过识别区块链数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎。

[0066] 步骤S800,将分片数据存储至目标分布式存储引擎中的存储节点。

[0067] 本实施例区块链数据的存储方式为分布式存储,是将区块链数据分散存储于多台独立的设备上,采用可扩展的系统结构,利用多台存储服务器分担存储负荷,它不但提高了系统的可靠性、可用性和存取效率,还易于扩展。存储节点即一台台独立的用于存储数据的设备(存储服务器)。当获取到对应的分布式存储引擎后,将分片数据存储至目标分布式存储引擎中存储节点。具体的,可以根据分片数据的哈希值,将分片数据存储至哈希值指向的存储节点。也可以是识别分片数据的数据量或者数据重要程度级别标识,根据数据量或数据重要程度级别标识,将分片数据分配至级别标识对应的各存储节点进行存储。

[0068] 上述区块链数据归档存储方法,通过周期性检测待归档数据并对其进行分片处理,能够避免数据冗余,降低运算速度,根据分片数据来源能够灵活的选择存储方式,保全数据的安全可用性,并且将分片数据分配至对应的分布式存储引擎中的存储节点能够节省节点存储空间,保证存储空间良好性以及数据高可用性,提高数据处理效率。

[0069] 如图3所示,在其中一个实施例中,周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据包括:步骤S220,按照周期时间统计区块链节点中最低高度值的数据与最高高度值的数据之间的数据总量,周期时间小于生成满足预设总量的数据集的所需时间,当数据总量大于预设总量时,则从最低高度值依次获取满足预设总量的数据集,检测数据集中每个数据的访问频率,当数据集中每个数据均小于预设访问频率时,则将数据集中的数据确定为待归档数据。

[0070] 具体可以是查看区块链节点本地存储的个数,即最低高度值的数据与最高高度值的数据之间的所有数据总量(即数据总个数),若数据总量大于预设总量,则从最低高度值的数据开始获取预设总量的数据,若每个预设总量的数据访问量均小于预设访频率时,确定预设总量为满足归档条件的数据,其中,周期时间小于达到预设总量的数据的生成时间。预设数据总量可以为 $n$ 个,具体可根据服务器的数据处理能力进行自由设置,生成预设总量的数据的所需时间由业务处理过程中数据流量可以确定。将检测的周期时间设置为小于预设总量的数据生成时间,才能在再次生成预设总量的数据前将已经满足预设总量的数据进行归档,避免满足归档条件的数据积累。具体的,例如本地存储的数据的最低高度值为 $H_0$ ,最高高度值数据为 $H_{31}$ ,则本地存储数据总量为 $H_{31}-H_0+1=31$ 个,即最低高度值到最高高度值之间有31个数据,当预设总量为20个时,本地存储的最低高度值的数据与最高高度值之间的数据总量(31个)大于预设总量(20个),则从最低高度值的数据开始获取满足该预设总量20个的数据,即 $H_0-H_{19}$ 的数据。进一步的,采用列表记录最近一段时间内访问频率超过预设访问频率的数据以及数据对应的高度值,预设访问频率为 $p$ ,访问频率超过 $p$ 的数据可以为 $\{X_1 \dots X_{30}\}$ ,当预设总量的数据 $H_0-H_{19}$ 均不在 $\{X_1 \dots X_{30}\}$ 时,确定该 $H_0-H_{19}$ 为待归档数据。本实施例中,将检测的周期时间设置为小于生成预设总量的数据的所需时间,能在再次生成预设总量的数据前将以满足预设总量的数据块进行数据归档,避免本地待归档数据的积累,造成数据冗余。

[0071] 如图3所示,在其中一个实施例中,对待归档数据进行分片处理,得到分片数据包括:步骤S420,采用一致性哈希算法对待归档数据进行分片处理,得到分片数据。

[0072] 以一致性哈希(Hash)算法为例,可采用一致性哈希的割环算法来实现数据分片,将哈希环切割为相同大小的分片,然后将这些分片交给不同的存储节点负责。具体的,通过哈希算法将对应的Key(键值,如图4所示的 $K_1$ )哈希到环形哈希空间中,将分组数据通过特

定的哈希函数计算得到数据对应的哈希值,将哈希值散列到哈希环(Hash环,如图4所示的圆环)上,形成分片数据,当一个存储节点退出时,其所负责的分片并不需要顺时针合并之后交给存储节点,如图4所述的存储节点a,b以及c,而是可以更灵活的将整个分片作为一个整体交给任意存储节点。在实践中,一个分片多作为最小的数据迁移和备份单位。可以理解的是,在其他实施例中,分片处理方法还可以是轮流放置或区间划分等方式进行。本方案中,数据分组至少形成3个数据分片,每个数据分片都由部分分组数据组成(不需要包括所有的分组数据),每个数据分片中的数据分组数量可以相同也可以不同,同样也需要保证每个数据分组至少添加至两个数据分片,形成至少两个存储副本,即使存储系统中某一个存储节点遭到攻击,也不会泄露完整的数据,能够根据其他存储节点的副本数据对丢失数据进行修复。通过对数据进行分片处理,提高数据处理速度和数据吞吐量,防止数据量过大造成阻塞,且一致性Hash可以很好的解决稳定性问题,可以将所有的存储节点排列在收尾相接的Hash环上。

[0073] 如图3所示,在其中一个实施例中,分布式存储引擎包括公有式存储引擎、联盟式存储引擎以及私有式存储引擎,识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎包括:步骤S620,当识别到分片数据来源于私有区块链时,修改通用私有式存储引擎对应的入口参数,获取私有式存储引擎,当识别到分片数据基于智能合约来源于联盟区块链时,修改通用联盟式存储引擎对应的入口参数,获取联盟式存储引擎,当识别到分片数据来源于公有式区块链时,修改通用公有式存储引擎对应的入口参数,获取公有式存储引擎。

[0074] 分布式存储引擎包括公有式存储引擎、联盟式存储引擎以及私有式存储引擎,数据来源类型可以包括私有区块链(私有链)、联盟区块链(联盟链)以及公有式区块链(公有链),私有式存储引擎、联盟式存储引擎以及公有式存储引擎分别指的是用于存储上述私有区块链的数据、联盟区块链的数据以及公有式区块链的数据的存储引擎。当完成对待归档数据的分片处理,得到分片数据后,识别分片数据的数据来源类型,获取与数据来源对应的分布式存储引擎。具体的,当识别分片数据来源于私有区块链时,修改通用私有式存储引擎对应的入口参数,获取私有式存储引擎,私有式存储引擎用于存储来源私有区块链的数据,供自身使用;当识别分片数据基于智能合约来源于联盟区块链时,修改通用联盟式存储引擎对应的入口参数,获取联盟式存储引擎,联盟式存储引擎用于存储来源于联盟区块链的数据,供特定的多方使用;当识别分片数据来源于公有式区块链时,修改通用公有式存储引擎对应的入口参数,获取公有式存储引擎,公有式存储引擎用于存储来源于公有式区块链的数据,供多方使用。也可以是接收参数更改请求,根据参数修改请求携带的目标修改参数,搭建与目标修改参数对应的存储引擎。本实施例中,根据分片数据的数据来源类型,灵活的选择存储方式,保证数据的安全可用性。

[0075] 如图3所示,在其中一个实施例中,将分片数据存储至目标分布式存储引擎中的存储节点包括:步骤S820,根据哈希算法,计算分片数据的哈希值,将分片数据存储至目标分布式存储引擎中哈希值指向的存储节点。

[0076] 当获取对应的分布式存储引擎后,可以是根据哈希算法计算每一个分片数据的哈希值,哈希值指向分片数据的存储地址,将分片数据存储至目标分布式存储引擎中哈希值指向的存储节点。以私有式存储引擎为例,根据分片数据的哈希值指向的存储地址,将分片

数据存储至私有式存储引擎中的存储地址对应的存储节点。在其他实施例中,也可以预设分片数据的预设重要程度级别,识别各个分片数据的重要等级标识,当分片数据的重要程度级别为高等级时,选择算力强、算速快的存储节点进行存储。也可以是预设分片数据的数据量,识别分片数据所具有的数据量,当分片数据所携带的数据量高于预设阈值时,将分片数据分配至容量大的存储节点。本实施例中,根据分片数据的哈希值完成分片数据的存储节点的分配,对应的存储分配数据,能够便于记录分配数据与存储节点的对应关系。

[0077] 在其中一个实施例中,将分片数据存储至目标分布式存储引擎中的存储节点包括:步骤S840,对各存储节点进行评估归类,确定各存储节点的存储性能,对分片数据进行权值分配评估,确定分片数据的存储需求,根据各存储节点的存储性能以及各分片数据的存储需求,将分片数据存储至目标分布式存储引擎中的存储节点。

[0078] 分片数据的存储方式还可以是预先对存储节点进行评估归类,具体的,根据存储节点的容量、算速以及算力(计算能力)等对存储节点做出评估,确定存储节点的存储性能;同时对分片数据的重要程度、数据量以及类型等进行权值分配评估,确定分片数据的存储需求,然后根据各候选节点的存储性能和各分片数据存储需求,进行对等选择,将分片数据存储至目标分布式存储引擎中合适的存储节点。其中,存储节点是各自独立工作的节点。具体的,可以是识别分片数据的数据保密程度、数据量以及数据类型等,根据各存储节点的算力、算速以及容量等性能对各存储节点进行编号,例如将算力最强、算速最快或容量最大的存储节点依次编号为1、2、3...m等,将重要数据或者数据量大或数据保密程度高的数据依次发送至编号为1、2、3...m的存储节点进行储存。本实施例中,通过对数据分片,以及将数据发送至对应的存储节点,实现数据的高吞吐,降低设备的技术成本。

[0079] 如图3所示,在其中一个实施例中,分片数据所在的数据分片包括多个数据分组;将分片数据存储至目标分布式存储引擎中的存储节点之后,还包括:步骤S900,计算各数据分组的哈希值,根据各数据分组的哈希值,构建与各数据分组所在的数据分片对应的梅克尔树,记录各梅克尔树与各数据分片所在的存储节点的对应关系。

[0080] 梅克尔树或称墨克树、哈希树,是一种树形数据结构,树的每个叶节点存储了数据块的哈希值。分片数据所在的数据分片包括多个数据分组,在计算得到各数据分组的哈希值后,可以各数据分组的哈希值为基础建立与各数据分组所在的数据分片对应的梅克尔树,为便于理解,可称作分片梅克尔树,每个数据分片对应一个分片梅克尔树,在获取到各数据分片对应的分片梅克尔树后,可以将各分片梅克尔树与各数据分片所在存储节点之间的对应关系进行记录。本实施例中,通过梅克尔树能够实现对数据的查找和验证,提高储存数据的私密性和安全性。

[0081] 如图3所示,在其中一个实施例中,还包括:步骤S950,对各存储节点进行容量检测,当各存储节点容量达到预设存储容量时,发送存储节点扩展请求。

[0082] 按照周期时间对各存储节点的容量进行检测,当各存储节点容量达到预设存储容量时,则发送存储节点扩展请求,具体可以是发送存储节点扩展请求至存储节点中的主节点,主节点在接收到存储节点扩展请求后,主动增加其他存储节点来达到扩展节点的目的。其中,主节点具备管理集群中的一些变更,例如新建或删除索引、增加或移除其他节点等。也可以是发送存储节点扩展请求至管理终端,管理终端接收存储节点扩展请求后,增加其他存储节点扩充存储容量。本实施例中,通过对存储节点定期进行容量检测,扩充存储空

间,能够避免存储节点的存储空间不足造成数据丢失的问题。

[0083] 应该理解的是,虽然图2-3的流程图中的各个步骤按照箭头的指示依次显示,但是这些步骤并不是必然按照箭头指示的顺序依次执行。除非本文中有明确的说明,这些步骤的执行并没有严格的顺序限制,这些步骤可以以其它的顺序执行。而且,图2-3中的至少一部分步骤可以包括多个子步骤或者多个阶段,这些子步骤或者阶段并不必然是在同一时刻执行完成,而是可以在不同的时刻执行,这些子步骤或者阶段的执行顺序也不必然是依次进行,而是可以与其它步骤或者其它步骤的子步骤或者阶段的至少一部分轮流或者交替地执行。

[0084] 在其中一个实施例中,如图5所示,提供了一种区块链数据归档存储装置,包括:数据检测模块410、数据分片模块420、分布式存储引擎获取模块430和数据存储模块440,其中:

[0085] 数据检测模块410,用于周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,预设归档条件包括高度值条件以及访问频率条件;

[0086] 数据分片模块420,用于对待归档数据进行分片处理,得到分片数据;

[0087] 分布式存储引擎获取模块430,用于识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎;

[0088] 数据存储模块440,用于将分片数据存储至目标分布式存储引擎中的存储节点。

[0089] 如图6所示,在其中一个实施例中,区块链数据归档存储装置还包括关系记录模块450,用于计算各数据分组的哈希值,根据各数据分组的哈希值,构建与各数据分组所在的数据分片对应的梅克尔树,记录各梅克尔树与各数据分片所在的存储节点的对应关系。

[0090] 如图6所示,在其中一个实施例中,区块链数据归档存储装置还包括容量检测模块460,用于对各存储节点进行容量检测,当各存储节点容量达到预设存储容量时,发送存储节点扩展请求。

[0091] 在其中一个实施例中,数据检测模块410还用于当数据总量大于预设总量时,则从最低高度值依次获取满足预设总量的数据集,检测数据集中每个数据的访问频率,当数据集中每个数据均小于预设访问频率时,则将数据集中的数据确定为待归档数据。

[0092] 在其中一个实施例中,分布式存储引擎获取模块430还用于当识别到分片数据来源于私有区块链时,修改通用私有式存储引擎对应的入口参数,获取私有式存储引擎,当识别到分片数据基于智能合约来源于联盟区块链时,修改通用联盟式存储引擎对应的入口参数,获取联盟式存储引擎,当识别到分片数据来源于公有式区块链时,修改通用公有式存储引擎对应的入口参数,获取目标公有式存储引擎。

[0093] 在其中一个实施例中,数据存储模块440还用于根据哈希算法,计算分片数据的哈希值,将分片数据存储至目标分布式存储引擎中哈希值指向的存储节点;或者,对各存储节点进行评估归类,确定各存储节点的存储性能,对分片数据进行权值分配评估,确定分片数据的存储需求,根据各存储节点的存储性能以及各分片数据的存储需求,将分片数据存储至目标分布式存储引擎中的存储节点。

[0094] 关于区块链数据归档存储装置的具体限定可以参见上文中对于区块链数据归档存储方法的限定,在此不再赘述。上述区块链数据归档存储装置中的各个模块可全部或部分通过软件、硬件及其组合来实现。上述各模块可以硬件形式内嵌于或独立于计算机设备

中的处理器中,也可以以软件形式存储于计算机设备中的存储器中,以便于处理器调用执行以上各个模块对应的操作。

[0095] 在其中一个实施例中,提供了一种计算机设备,该计算机设备可以是服务器,其内部结构图可以如图7所示。该计算机设备包括通过系统总线连接的处理器、存储器、网络接口和数据库。其中,该计算机设备的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统、计算机程序和数据库。该内存储器为非易失性存储介质中的操作系统和计算机程序的运行提供环境。该计算机设备的数据库用于存储区块链数据。该计算机设备的网络接口用于与外部的终端通过网络连接通信。该计算机程序被处理器执行时以实现一种区块链数据归档存储方法。

[0096] 本领域技术人员可以理解,图7中示出的结构,仅仅是与本申请方案相关的部分结构的框图,并不构成对本申请方案所应用于其上的计算机设备的限定,具体的计算机设备可以包括比图中所示更多或更少的部件,或者组合某些部件,或者具有不同的部件布置。

[0097] 在其中一个实施例中,提供了一种计算机设备,包括存储器和处理器,存储器中存储有计算机程序,该处理器执行计算机程序时实现以下步骤:周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,预设归档条件包括高度值条件以及访问频率条件,对待归档数据进行分片处理,得到分片数据,识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎,将分片数据存储至目标分布式存储引擎中的存储节点。

[0098] 在其中一个实施例中,处理器执行计算机程序时还实现以下步骤:按照周期时间统计区块链节点中最低高度值的数据与最高高度值的数据之间的数据总量,周期时间小于生成满足预设总量的数据集的所需时间,当数据总量大于预设总量时,则从最低高度值依次获取满足预设总量的数据集,检测数据集中每个数据的访问频率,当数据集中每个数据均小于预设访问频率时,则将数据集中的数据确定为待归档数据。

[0099] 在其中一个实施例中,处理器执行计算机程序时还实现以下步骤:当识别到分片数据来源于私有区块链时,修改通用私有式存储引擎对应的入口参数,获取私有式存储引擎,当识别到分片数据基于智能合约来源于联盟区块链时,修改通用联盟式存储引擎对应的入口参数,获取联盟式存储引擎,当识别到分片数据来源于公有式区块链时,修改通用公有式存储引擎对应的入口参数,获取目标公有式存储引擎。

[0100] 在其中一个实施例中,处理器执行计算机程序时还实现以下步骤:根据哈希算法,计算分片数据的哈希值,将分片数据存储至目标分布式存储引擎中哈希值指向的存储节点。

[0101] 在其中一个实施例中,处理器执行计算机程序时还实现以下步骤:对各存储节点进行评估归类,确定各存储节点的存储性能,对分片数据进行权值分配评估,确定分片数据的存储需求,根据各存储节点的存储性能以及各分片数据的存储需求,将分片数据存储至目标分布式存储引擎中的存储节点。

[0102] 在其中一个实施例中,处理器执行计算机程序时还实现以下步骤:计算各数据分组的哈希值,根据各数据分组的哈希值,构建与各数据分组所在的数据分片对应的梅克尔树,记录各梅克尔树与各数据分片所在的存储节点的对应关系。

[0103] 在其中一个实施例中,处理器执行计算机程序时还实现以下步骤:对各存储节点

进行容量检测,当各存储节点容量达到预设存储容量时,发送存储节点扩展请求。

[0104] 在其中一个实施例中,提供了一种计算机可读存储介质,其上存储有计算机程序,计算机程序被处理器执行时实现以下步骤:周期性检测区块链数据中满足预设归档条件的数据,得到待归档数据,预设归档条件包括高度值条件以及访问频率条件,对待归档数据进行分片处理,得到分片数据,识别分片数据的数据来源类型,获取与数据来源类型对应的分布式存储引擎,得到目标分布式存储引擎,将分片数据存储至目标分布式存储引擎中的存储节点。

[0105] 在其中一个实施例中,计算机程序被处理器执行时还实现以下步骤:按照周期时间统计区块链节点中最低高度值的数据与最高高度值的数据之间的数据总量,周期时间小于生成满足预设总量的数据集的所需时间,当数据总量大于预设总量时,则从最低高度值依次获取满足预设总量的数据集,检测数据集中每个数据的访问频率,当数据集中每个数据均小于预设访问频率时,则将数据集中的数据确定为待归档数据。

[0106] 在其中一个实施例中,计算机程序被处理器执行时还实现以下步骤:当识别到分片数据来源于私有区块链时,修改通用私有式存储引擎对应的入口参数,获取私有式存储引擎,当识别到分片数据基于智能合约来源于联盟区块链时,修改通用联盟式存储引擎对应的入口参数,获取联盟式存储引擎,当识别到分片数据来源于公有式区块链时,修改通用公有式存储引擎对应的入口参数,获取目标公有式存储引擎。

[0107] 在其中一个实施例中,计算机程序被处理器执行时还实现以下步骤:根据哈希算法,计算分片数据的哈希值,将分片数据存储至目标分布式存储引擎中哈希值指向的存储节点。

[0108] 在其中一个实施例中,计算机程序被处理器执行时还实现以下步骤:对各存储节点进行评估归类,确定各存储节点的存储性能,对分片数据进行权值分配评估,确定分片数据的存储需求,根据各存储节点的存储性能以及各分片数据的存储需求,将分片数据存储至目标分布式存储引擎中的存储节点。

[0109] 在其中一个实施例中,计算机程序被处理器执行时还实现以下步骤:计算各数据分组的哈希值,根据各数据分组的哈希值,构建与各数据分组所在的数据分片对应的梅克尔树,记录各梅克尔树与各数据分片所在的存储节点的对应关系。

[0110] 在其中一个实施例中,计算机程序被处理器执行时还实现以下步骤:对各存储节点进行容量检测,当各存储节点容量达到预设存储容量时,发送存储节点扩展请求。

[0111] 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,所述的计算机程序可存储于一非易失性计算机可读取存储介质中,该计算机程序在执行时,可包括如上述各方法的实施例的流程。其中,本申请所提供的各实施例中所使用的对存储器、存储、数据库或其它介质的任何引用,均可包括非易失性和/或易失性存储器。非易失性存储器可包括只读存储器(ROM)、可编程ROM(PROM)、电可编程ROM(EPROM)、电可擦除可编程ROM(EEPROM)或闪存。易失性存储器可包括随机存取存储器(RAM)或者外部高速缓冲存储器。作为说明而非局限,RAM以多种形式可得,诸如静态RAM(SRAM)、动态RAM(DRAM)、同步DRAM(SDRAM)、双数据率SDRAM(DDRSDRAM)、增强型SDRAM(ESDRAM)、同步链路(Synchlink)DRAM(SLDRAM)、存储器总线(Rambus)直接RAM(RDRAM)、直接存储器总线动态RAM(DRDRAM)、以及存储器总线动态RAM(RDRAM)等。

[0112] 以上实施例的各技术特征可以进行任意的组合,为使描述简洁,未对上述实施例中的各个技术特征所有可能的组合都进行描述,然而,只要这些技术特征的组合不存在矛盾,都应当认为是本说明书记载的范围。

[0113] 以上所述实施例仅表达了本申请的几种实施方式,其描述较为具体和详细,但不能因此而理解为对发明专利范围的限制。应当指出的是,对于本领域的普通技术人员来说,在不脱离本申请构思的前提下,还可以做出若干变形和改进,这些都属于本申请的保护范围。因此,本申请专利的保护范围应以所附权利要求为准。

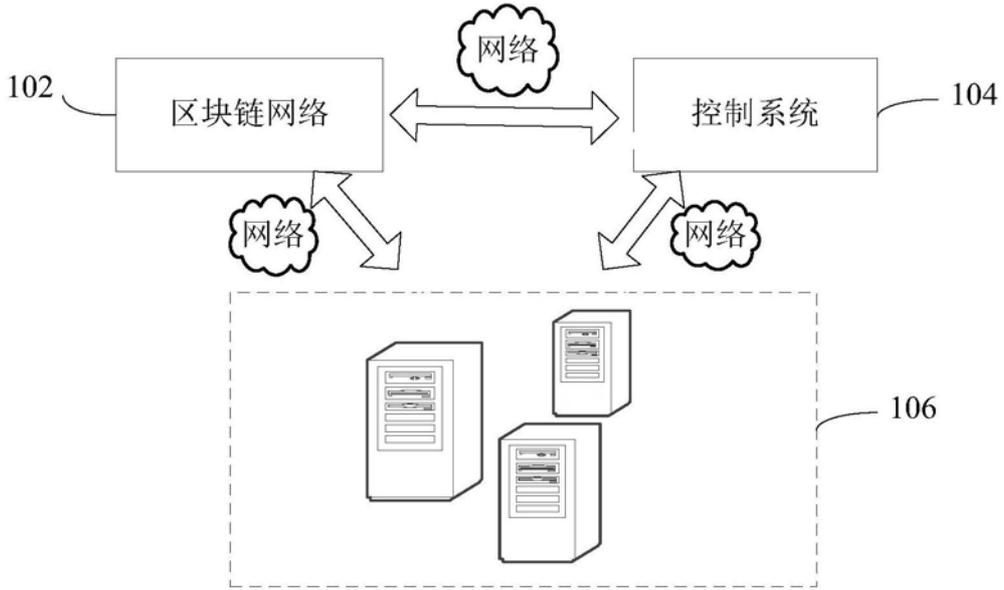


图1

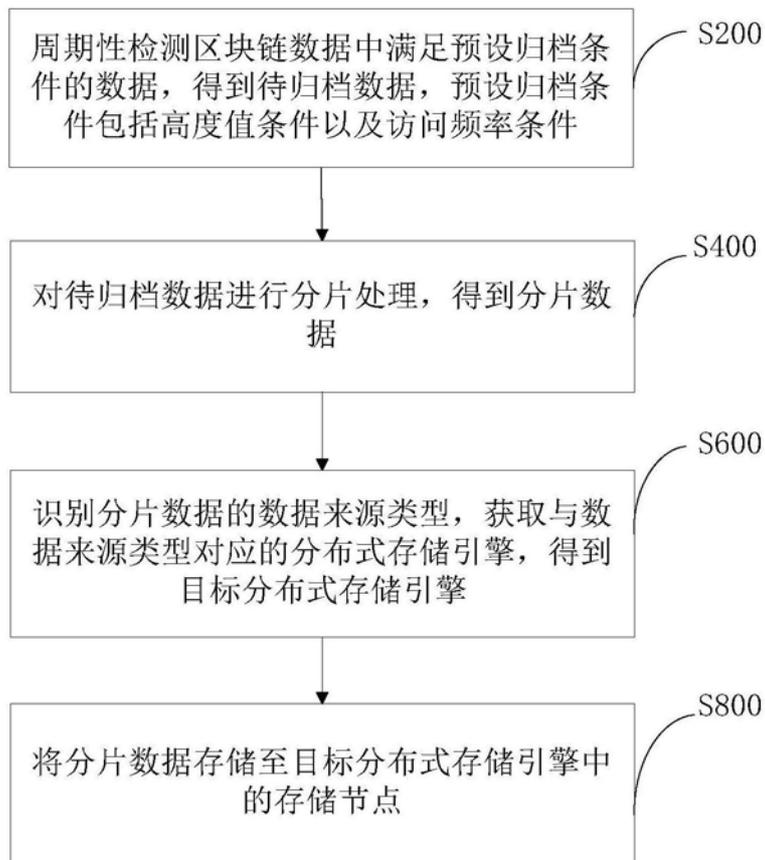


图2

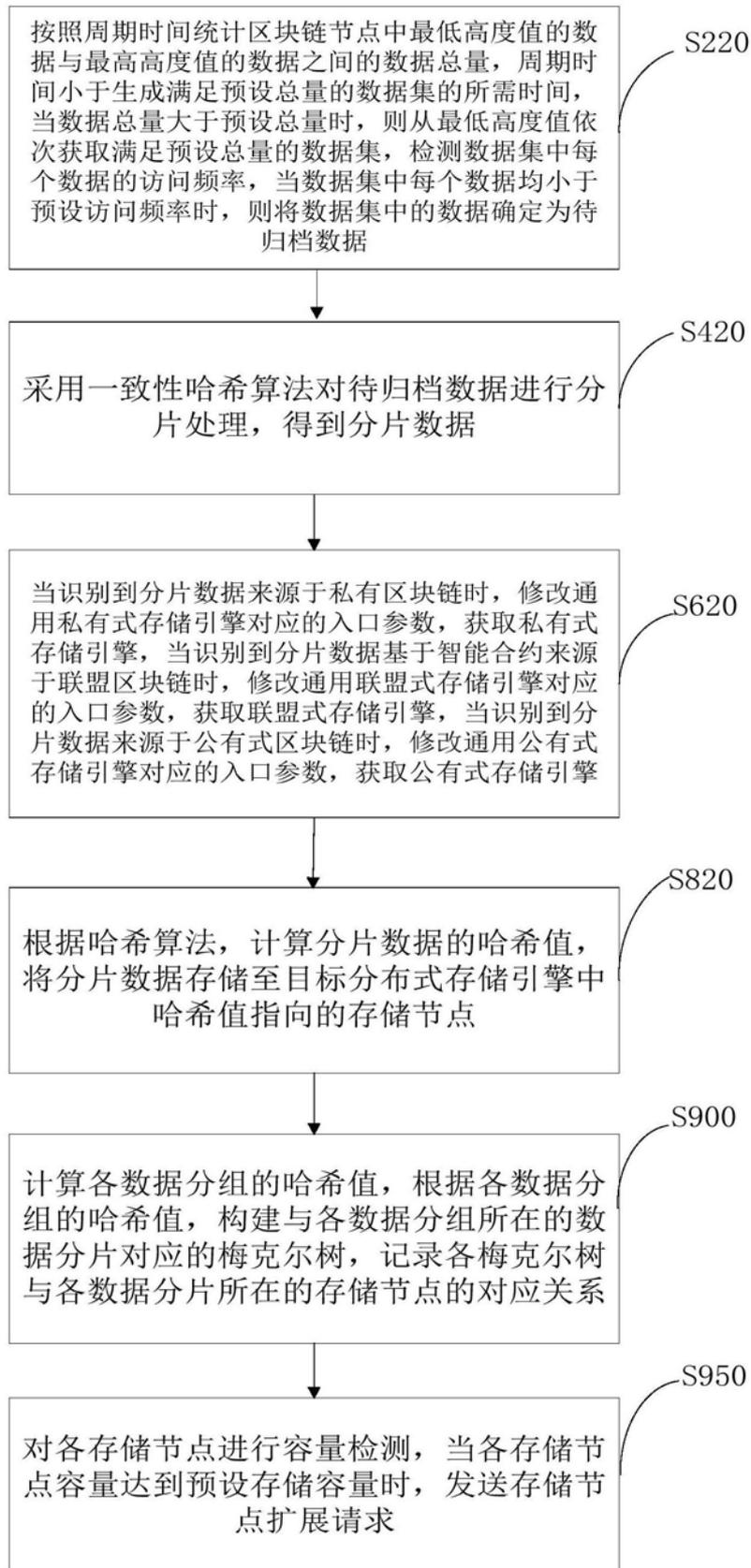


图3

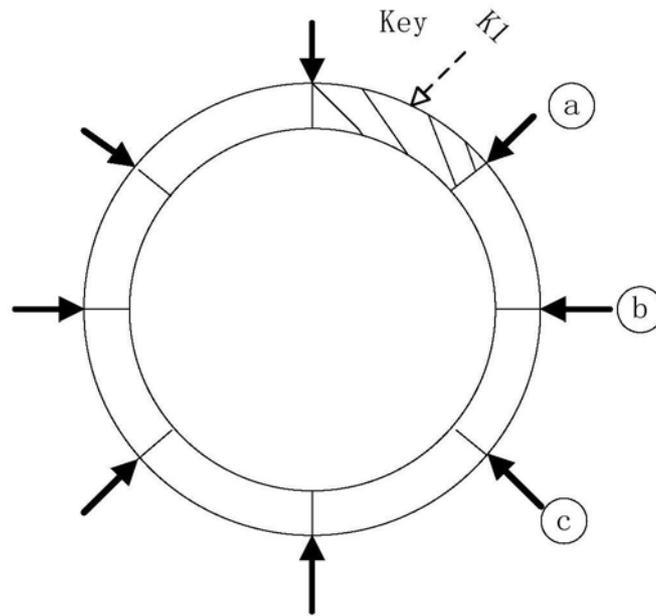


图4

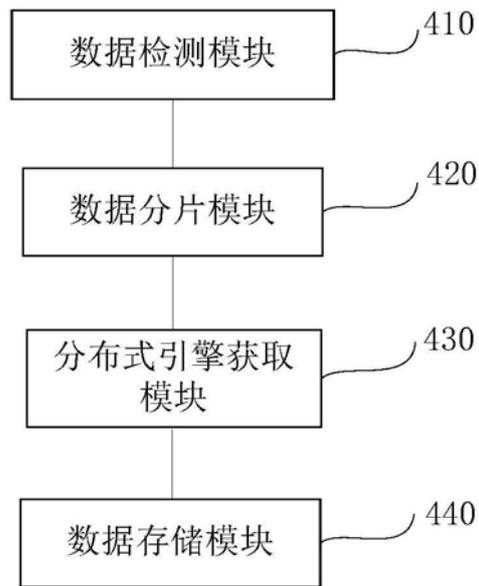


图5

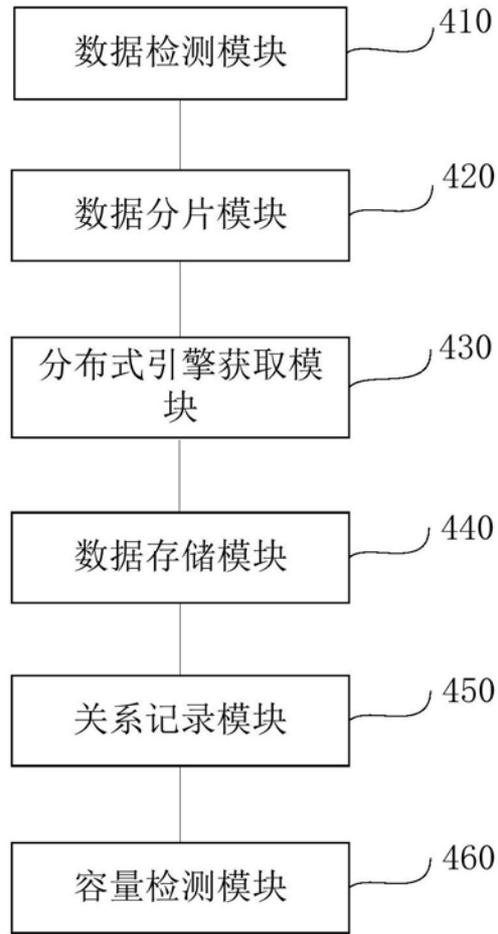


图6

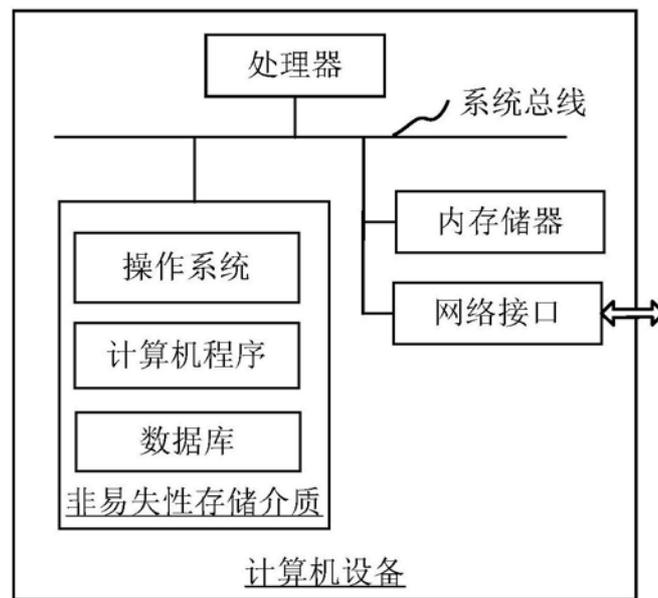


图7