



(12)发明专利申请

(10)申请公布号 CN 111309720 A

(43)申请公布日 2020.06.19

(21)申请号 201811509734.7

(22)申请日 2018.12.11

(71)申请人 北京京东尚科信息技术有限公司
地址 100080 北京市海淀区知春路76号8层
申请人 北京京东世纪贸易有限公司

(72)发明人 张洪斌

(74)专利代理机构 北京品源专利代理有限公司
11332

代理人 孟金喆

(51)Int.Cl.

G06F 16/22(2019.01)

G06F 16/2458(2019.01)

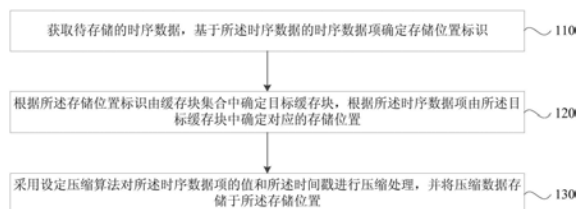
权利要求书3页 说明书15页 附图8页

(54)发明名称

时序数据的存储、读取方法、装置、电子设备
及存储介质

(57)摘要

本发明实施例公开了时序数据的存储、读取方法、装置、电子设备及存储介质。该时序数据的存储方法包括获取待存储的时序数据,基于该时序数据的时序数据项确定存储位置标识,其中,该时序数据包含时序数据项、时序数据项的值和时间戳;根据存储位置标识由缓存块集合中确定目标缓存块,根据该时序数据项由目标缓存块中确定对应的存储位置;采用设定压缩算法对时序数据项的值和时间戳进行压缩处理,并将压缩数据存储于存储位置。本发明实施例通过采用上述技术方案,克服了时序数据大量占用设备的存储空间,提高了空间利用率;此外,由于缓存的是与时间关联的压缩数据,可以避免发生因多次执行查询指令导致的查询延迟问题,降低了查询延时。



1. 一种时序数据的存储方法,其特征在于,包括:

获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识,其中,所述时序数据包含所述时序数据项、时序数据项的值和时间戳;

根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

采用设定压缩算法对所述时序数据项的值和所述时间戳进行压缩处理,并将压缩数据存储于所述存储位置。

2. 根据权利要求1所述的方法,其特征在于,获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识,包括:

获取待存储的时序数据的时序数据项,根据哈希函数计算所述时序数据项对应的哈希值;

根据缓存块集合中的缓存块的数量确定缓存节点数量,对所述哈希值和缓存节点数量进行取模运算,得到缓存节点标识,其中,所述缓存节点数量与所述缓存块的数量存在对数关系;

基于所述缓存节点标识与缓存块的对应关系,确定用于存储所述时序数据的缓存块,将所述缓存块的标识信息作为存储位置标识。

3. 根据权利要求1或2所述的方法,其特征在于,采用设定压缩算法对所述时序数据项的值和所述时间戳进行压缩处理,并将压缩数据存储于所述存储位置,包括:

获取所述待存储的时序数据中的时间戳和时序数据项的值;

根据所述时间戳、首个时序数据中的时间戳与压缩序列中已写入的最后一个压缩数据的时间戳压缩值之间的差值、以及所述时序数据项的值与首个时序数据中的时序数据项的值之间的相似度,对所述待存储的时序数据中的时间戳和时序数据项的值进行无损压缩处理,得到压缩数据;

将所述压缩数据存储于所述存储位置上的压缩序列中;其中,所述首个时序数据是指所述压缩序列最近一次被清空后所述压缩序列中被存储的第一个时序数据并且首个时序数据未被压缩。

4. 根据权利要求3所述的方法,其特征在于,根据所述时间戳、首个时序数据中的时间戳与压缩序列中已写入的最后一个压缩数据的时间戳压缩值之间的差值、以及所述时序数据项的值与首个时序数据中的时序数据项的值之间的相似度,对所述待存储的时序数据中的时间戳和时序数据项的值进行无损压缩处理,得到压缩数据,包括:

计算所述时间戳与压缩序列中首个时序数据的时间戳之间的第一差值,计算所述第一差值与所述压缩序列中已写入的最后一条压缩数据的时间戳压缩值之间的第二差值;

计算所述时序数据项的值与首个时序数据中的时序数据项的值的异或运算结果,根据所述异或运算结果匹配预设压缩策略;

基于所述预设压缩策略对所述待存储的时序数据中的时序数据项的值进行无损压缩,将无损压缩结果与所述第二差值的组合作为压缩数据。

5. 根据权利要求4所述的方法,其特征在于,所述预设压缩策略包括:

若异或运算结果为0,则确定所述待存储的时序数据中的时序数据项的值的压缩值为0;

若异或运算结果为非0,则根据所述异或运算结果中的每个非0值以及与所述非0值相邻的0的数量确定所述待存储的时序数据中的时序数据项的值的压缩值。

6. 根据权利要求3所述的方法,其特征在于,在获取所述待存储的时序数据中的时间戳之后,还包括:

获取所述压缩序列对应的开始时间戳和结尾时间戳,其中,所述开始时间戳为所述压缩序列内已存储的时序数据中最小的时间戳,所述结尾时间戳为所述压缩序列内已存储的时序数据中最大的时间戳;

将所述待存储的时序数据中的时间戳与所述开始时间戳和所述结尾时间戳进行比较,根据比较结果更新所述压缩序列对应的开始时间戳或结尾时间戳。

7. 根据权利要求6所述的方法,其特征在于,还包括:

按照设定周期将所述压缩序列中的所述压缩数据导出至设定的压缩数组,以清空所述压缩序列;

将所述压缩序列的所述开始时间戳和结尾时间戳存储至所述压缩数组。

8. 一种时序数据的读取方法,其特征在于,包括:

获取时序数据项和目标时间的时间戳,基于所述时序数据项确定待读取的时序数据的存储位置标识;

根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

根据所述目标时间的时间戳由所述存储位置读取压缩数据;

采用设定解压缩算法对所述压缩数据进行解压得到待读取的时序数据。

9. 根据权利要求8所述的方法,其特征在于,基于所述时序数据项确定待读取的时序数据的存储位置标识,包括:

根据哈希函数计算所述时序数据项对应的哈希值;

根据缓存块集合中的缓存块的数量确定缓存节点数量,对所述哈希值和缓存节点数量进行取模运算,得到缓存节点标识,其中,所述缓存节点数量与所述缓存块的数量存在对数关系;

基于所述缓存节点标识与缓存块的对应关系,确定存储待读取的时序数据的缓存块,将所述缓存块的标识信息作为存储位置标识。

10. 根据权利要求8或9所述的方法,其特征在于,根据所述目标时间的时间戳由所述存储位置读取压缩数据,包括:

获取所述存储位置上的压缩序列对应的开始时间戳和结尾时间戳,其中,所述开始时间戳为所述压缩序列内已存储的时序数据中最小的时间戳,所述结尾时间戳为所述压缩序列内已存储的时序数据中最大的时间戳;

若所述目标时间的时间戳属于由所述开始时间戳和结尾时间戳构成的时间戳区间,则由所述压缩序列中获取已存储的压缩数据;

若所述目标时间的时间戳不属于所述时间戳区间,则根据所述目标时间的时间戳匹配设定压缩数组内的开始时间戳和结尾时间戳,获取匹配成功的设定压缩数组内的压缩数据。

11. 根据权利要求10所述的方法,其特征在于,采用设定解压缩算法对所述压缩数据进

行解压得到待读取的时序数据,包括:

获取所述压缩序列或匹配成功的设定压缩数组中存储的首个时序数据的时间戳和时序数据项的值;

获取所述压缩数据中的待解压的时序数据的时间戳压缩值,计算所述时间戳压缩值、首个时序数据的时间戳与待解压的时序数据的上一时间相邻的时序数据的时间戳压缩值的和值,得到所述待解压的时序数据中的时间戳;

获取所述压缩数据中的待解压的时序数据的时序数据项的值的压缩值,计算所述压缩值与首个时序数据中的时序数据项的值的异或运算结果,得到所述待解压的时序数据中的时序数据项的值;

将所述目标时间的时间戳与解压后的时序数据中的时间戳进行匹配,根据匹配结果由解压后的时序数据中确定待读取的时序数据。

12. 一种时序数据的存储装置,其特征在于,包括:

存储位置标识确定模块,用于获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识,其中,所述时序数据包含所述时序数据项、时序数据项的值和时间戳;

存储位置确定模块,用于根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

数据存储模块,用于采用设定压缩算法对所述时序数据项的值和所述时间戳进行压缩处理,并将压缩数据存储于所述存储位置。

13. 一种时序数据的读取装置,其特征在于,包括:

读取位置标识确定模块,用于获取时序数据项和目标时间,基于所述时序数据项确定待读取的时序数据的读取位置标识;

读取位置确定模块,用于根据所述读取位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的读取位置;

压缩数据读取模块,用于根据所述目标时间的时间戳由所述存储位置读取压缩数据;

数据解压模块,用于采用设定解压缩算法对所述压缩数据进行解压得到待读取的时序数据。

14. 一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求1-7所述的时序数据的存储方法。

15. 一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求8-11所述的时序数据的读取方法。

16. 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如权利要求1-7所述的时序数据的存储方法。

17. 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如权利要求8-11所述的时序数据的读取方法。

时序数据的存储、读取方法、装置、电子设备及存储介质

技术领域

[0001] 本发明实施例涉及数据库技术领域,尤其涉及一种时序数据的存储、读取方法、装置、电子设备及存储介质。

背景技术

[0002] 随着互联网或者物联网的应用越来越广泛,互联网中的监控系统或者物联网设备等,产生了越来越多的基于时间序列的数据,这些数据被称为时序数据。

[0003] 相关技术中,通常将数小时之内的热点数据存储在内存中进行缓存。例如通过键值对的方式,以时序数据项为键,对时序数据项的值进行缓存。对于其它需要持久保存的时序数据,则存放在OpenTSDB (Open time series data base,开发时间序列数据库)、Hbase (Hadoop Database,分布式数据库)、InfluxDB (时序性数据库) 或者Elasticsearch (非关系型数据库) 等数据库中。

[0004] 在实现本发明过程中,发明人发现相关技术中的数据存储方案至少存在如下问题:第一、对于互联网中的监控系统或者物联网设备,每天大约产生上亿条数据,采用相关技术中的存储方案会大量占用设备的存储空间,提高了数据的存储成本。第二、采用键值对的方式存储时序数据,在进行数据查询时,每次执行查询指令返回一条记录。而实际场景中,用户关注的通常是一段时间内的多条记录,就需要执行多次查询指令,从而,导致查询延时较高。

发明内容

[0005] 本发明实施例提供一种时序数据的存储、读取方法、装置、电子设备及存储介质,可以降低时序数据占用的存储空间,提高查询效率。

[0006] 第一方面,本发明实施例提供了一种时序数据的存储方法,包括:

[0007] 获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识,其中,所述时序数据包含所述时序数据项、时序数据项的值和时间戳;

[0008] 根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

[0009] 采用设定压缩算法对所述时序数据项的值和所述时间戳进行压缩处理,并将压缩数据存储于所述存储位置。

[0010] 第二方面,本发明实施例还提供了一种时序数据的读取方法,包括:

[0011] 获取时序数据项和目标时间的时间戳,基于所述时序数据项确定待读取的时序数据的存储位置标识;

[0012] 根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

[0013] 根据所述目标时间的时间戳由所述存储位置读取压缩数据;

[0014] 采用设定解压缩算法对所述压缩数据进行解压得到待读取的时序数据。

[0015] 第三方面,本发明实施例还提供了一种时序数据的存储装置,该装置包括:

[0016] 存储位置标识确定模块,用于获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识,其中,所述时序数据包含所述时序数据项、时序数据项的值和时间戳;

[0017] 存储位置确定模块,用于根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

[0018] 数据存储模块,用于采用设定压缩算法对所述时序数据项的值和所述时间戳进行压缩处理,并将压缩数据存储于所述存储位置。

[0019] 第四方面,本发明实施例还提供了一种时序数据的读取装置,该装置包括:

[0020] 读取位置标识确定模块,用于获取时序数据项和目标时间,基于所述时序数据项确定待读取的时序数据的读取位置标识;

[0021] 读取位置确定模块,用于根据所述读取位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的读取位置;

[0022] 压缩数据读取模块,用于根据所述目标时间的时间戳由所述存储位置读取压缩数据;

[0023] 数据解压模块,用于采用设定解压缩算法对所述压缩数据进行解压得到待读取的时序数据。

[0024] 第五方面,本发明实施例还提供了一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现如上述第一方面所述的时序数据的存储方法。

[0025] 第六方面,本发明实施例还提供了一种电子设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现如上述第二方面所述的时序数据的读取方法。

[0026] 第七方面,本发明实施例还提供了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如上述第一方面所述的时序数据的存储方法。

[0027] 第八方面,本发明实施例还提供了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如上述第二所述的时序数据的读取方法。

[0028] 本发明实施例通过对待存储的时序数据包含的时序数据项的值和时间戳进行压缩的方式,得到与时间关联的压缩数据,并将该压缩数据存储于时序数据项对应的目标缓存块中,克服了时序数据大量占用设备的存储空间的缺陷,提高了空间利用率;此外,由于缓存的是与时间关联的压缩数据,在对时序数据进行查询时,可以通过执行一次查询指令获得一段时间内的多条时序数据记录,避免发生因多次执行查询指令导致的查询延迟问题,降低了查询延时。

附图说明

[0029] 图1是本发明实施例提供的一种时序数据的存储方法的流程图;

[0030] 图2是本发明实施例提供的一种时序数据的分布式存储框架图;

[0031] 图3是本发明实施例提供的一种缓存服务器的缓存架构框图;

[0032] 图4是本发明实施例提供的一种时序数据的存储形式示意图;

- [0033] 图5是本发明实施例提供的另一种时序数据的存储方法的流程图；
- [0034] 图6是本发明实施例提供的另一种时序数据的存储方法的流程图；
- [0035] 图7是本发明实施例提供的一种时序数据的读取方法的流程图；
- [0036] 图8是本发明实施例提供的一种时序数据的存储装置的结构框图；
- [0037] 图9是本发明实施例提供的一种时序数据的读取装置的结构框图；
- [0038] 图10是本发明实施例提供的一种电子设备的结构框图；
- [0039] 图11是本发明实施例提供的另一种电子设备的结构框图。

具体实施方式

[0040] 下面结合附图和实施例对本发明作进一步的详细说明。可以理解的是，此处所描述的具体实施例仅仅用于解释本发明，而非对本发明的限定。另外还需要说明的是，为了便于描述，附图中仅示出了与本发明相关的部分而非全部结构。

[0041] 图1是本发明实施例提供的一种时序数据的存储方法的流程图，该方法可以由时序数据的存储装置来执行，其中，该装置可以由软件和/或硬件实现，一般可集成在电子设备中。如图1所示，该方法包括：

[0042] 步骤110、获取待存储的时序数据，基于所述时序数据的时序数据项确定存储位置标识。

[0043] 需要说明的是，时序数据是按照时间顺序记录的时间序列，包括时序数据项、时序数据项的值和时间戳。在对时序数据进行缓存时，可以选用Redis或者Memcache方式实现将时序数据存储于缓存服务器中。其中，Redis是一个高性能的key-value（键值对）数据库。MemCache是一个分布式内存对象缓存系统，在内存中对任意的数据（比如字符串、对象等）使用key-value形式进行存储，数据可以来自数据库调用、API（Application Programming Interface，应用程序编程接口）调用，或者页面渲染的结果等。

[0044] 本发明实施例中缓存服务器的缓存空间可以被分为多个缓存块，也就是说可以采用缓存块集合代表上述缓存空间。需要说明的是，各个缓存块之间不进行任何通信，在处理时序数据的读写业务时，根据接收到的读/写请求，采用一致性哈希算法进行读写操作。其中，一致性哈希算法的实现原理包括：构造一个设定长度的哈希环，根据节点名称的哈希值将服务器节点放置于上述哈希环上；计算数据的键的哈希值，在上述哈希环上顺时针查找距离该键的哈希值最近的服务器节点，完成数据的键到服务器的映射查找。图2是本发明实施例提供的一种时序数据的分布式存储框架图。如图2所示，一台服务器210的缓存空间可以被分割为4块。每一块是一个缓存块220，从而实现将获取到的整块数据分片存储于不同的缓存块中。在接收到外部系统的写数据请求时，可以通过一致性哈希算法将待写入的数据存储于对应的缓存块220。图3是本发明实施例提供的一种缓存服务器的缓存架构框图。如图3所示，缓存服务器300（CacheServer）包括位置管理（slotmanager）模块310、远程调用（RpcServer）模块320以及缓存块集合（SharedMap）330。其中，位置管理模块310用于管理存储位置标识与缓存块340的关联关系。远程调用模块320用于相应远程调用请求。缓存块集合330包括N个缓存块340，N可以根据输入或输出时序数据的数据量进行配置。N的值越大，表示将缓存分成的块就越多，每个缓存块上存储的数据就越少，可以便于查询数据，但是，N值设置的过大，会导致读取数据时获取到较多的压缩数据，对于数据解压缩会耗费更多的

时间,从而导致读取延迟,因此需要根据实际情况合理的设置N的取值。需要说明的是,每个缓存块340中均具有日志单元341、数据清除单元342和持久化单元343。其中,日志单元341用于写wal (Write-Ahead Logging,预写式日志,在数据写入到数据库之前,先写入到日志,再将日志记录变更到存储器中),防止内存中的数据丢失,实现在缓存服务器掉电开机后将数据加载回缓存服务器。数据清除单元342用于对缓存块中的垃圾数据进行清理。持久化单元343,用于对缓存块上的压缩数据进行持久化处理。

[0045] 示例性的,本发明一个实施例中可以通过一致性哈希算法确定用于存储待存储时序数据的缓存块,实现方式可以是获取待存储的时序数据的时序数据项,记为key,根据哈希函数计算所述时序数据项对应的哈希值,将哈希值记为hash_code。根据缓存块集合中的缓存块的数量N确定缓存节点数量,将缓存节点数量记为num_slot,对哈希值和缓存节点数量进行取模运算,得到缓存节点标识,记为slot_id,即 $slot_id = hash_code \% num_slot$ 。其中,所述缓存节点数量与所述缓存块的数量存在对数关系,即 $num_slot = 2^N$ 。基于缓存节点标识与缓存块的对应关系,确定用于存储所述时序数据的缓存块。缓存节点标识于缓存块的对应关系可以是将缓存节点标识作为键,采用键值对的方式保存缓存节点标识与缓存块的标识信息。其中,缓存块的标识信息可以是缓存块地址或缓存块名称等等。可以将上述所确定缓存块的标识信息作为存储位置标识。

[0046] 步骤120、根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置。

[0047] 需要说明的是,缓存块上可能存储不同的时序数据项,需要在各个缓存块中预先设置了时序数据项与存储位置的关联关系,以对时序数据进行分类存储,例如,以时序数据项为键,采用缓存块哈希表保存存储位置的标识信息。缓存块集合中包含多个缓存块,可以实现对时序数据进行分片处理,分别对每片数据进行垃圾数据清理,避免因垃圾数据过多而对服务性能产生影响;此外,由于对时序数据进行分片,可以对每片数据加锁,实现适当的减小锁的粒度(也可以成为锁的范围),提升了读写性能。

[0048] 需要说明的是,存储位置是用来保存时序数据项对应的时序数据项的值和时间戳的内存空间。时序数据在存储位置中的存储形式可有很多种,本申请实施例并不在具体限定。例如,可以采用跳表、压缩序列和压缩数组等形式。跳表用于记录时序数据项相同的不同时序数据中时序数据项的值和时间戳。跳表中的每一项的实现方式可以是压缩序列。跳表中的每一项的实现方式还可以是压缩序列结合压缩数组,实现对时序数据的再次分块,便于在读取数据时快速查找到对应的目标数据。

[0049] 图4是本发明实施例提供的一种时序数据的存储形式示意图,如图4所示,缓存块集合410中的每个缓存块420上维护一个缓存哈希表430,该缓存哈希表430中存储有时序数据项与存储位置的关联关系。该缓存哈希表430的核心数据结构包括跳表440,跳表440的每项存储相同时序数据项的时序数据中的时序数据项的值和时间戳,存储形式可以是压缩序列450和压缩数组460。

[0050] 本发明实施例中,由于存储位置标识代表的是缓存块的标识信息,因此,基于存储位置标识查询缓存块集合可以确定与存储位置标识对应的目标缓存块。目标缓存块中存储有时序数据项与存储位置的关联关系,基于待存储的时序数据中的时序数据项以及上述关联关系,可以确定待存储的时序数据对应的存储位置。

[0051] 步骤130、采用设定压缩算法对所述时序数据项的值和所述时间戳进行压缩处理，并将压缩数据存储于所述存储位置。

[0052] 需要说明的是，设定压缩算法是对时间戳和时序数据项的值进行压缩的算法，实现将时序数据项的值存储为与时间关联的压缩数据块。本发明实施例中，设定压缩算法是基于待存储的时序数据中时间戳、首个时序数据中的时间戳与压缩序列中待存储的时序数据的上一相邻时序数据的时间戳压缩值之间的差值、以及待存储的时序数据中时序数据项的值与首个时序数据中的时序数据项的值之间的相似度，对待存储的时序数据中的时间戳和时序数据项的值进行无损压缩的算法。

[0053] 示例性的，获取待存储的时序数据中的时间戳和时序数据项的值；根据该时间戳、首个时序数据中的时间戳与压缩序列中待存储的时序数据的上一相邻时序数据的时间戳压缩值之间的差值、以及该时序数据项的值与首个时序数据中的时序数据项的值之间的相似度，对待存储的时序数据中的时间戳和时序数据项的值进行无损压缩处理，得到压缩数据；将该压缩数据存储于所确定的存储位置上的压缩序列中；其中，首个时序数据是指压缩序列最近一次被清空后该压缩序列中被存储的第一个时序数据并且首个时序数据未被压缩（也可以认为其时间戳压缩值就是该时序数据的时间戳）。需要说明的是，若待存储的时序数据是与第一时序数据的接收时间相邻的第二时序数据，则计算该第二时序数据中时间戳和首个时序数据中的时间戳之间的差值，再计算该差值与首个时序数据中的时间戳之间的差值。可选的，若待存储的时序数据是与第一时序数据的接收时间相邻的第二时序数据，则压缩序列中已写入的最后一个压缩数据的时间戳压缩值取0。需要说明的是，在每条时序数据被写入压缩序列之前，判断压缩序列是否将被清空，若是，则将该条待写入的时序数据作为第一个时序数据，并将该条待写入的时序数据中的时间戳和时序数据项的值存储于清空后的压缩序列中。后续的待写入的时序数据的压缩数据可以采用追加的方式写入该压缩序列。本发明实施例中的压缩算法实现对时序数据的无损压缩，提高了存储空间利用率，降低了时序数据的存储成本。根据对线上系统的数据缓存进行采集并分析发现，采用本发明实施例的压缩算法可以将存储空间的利用率提高约10倍左右。

[0054] 本发明实施例的技术方案，通过采用设定压缩算法对待存储的时序数据包含的时序数据项的值和时间戳进行压缩的方式，得到与时间关联的压缩数据，并将该压缩数据存储于时序数据项对应的目标缓存块中，克服了时序数据大量占用设备的存储空间的缺陷，提高了空间利用率；此外，由于缓存的是与时间关联的压缩数据，在对时序数据进行查询时，可以通过执行一次查询指令获得一段时间内的多条时序数据记录，避免发生因多次执行查询指令导致的查询延迟问题，降低了查询延时。

[0055] 在上述技术方案的基础上，进一步增加了按照设定周期述压缩序列中的压缩数据导出至设定的压缩数组，以清空压缩序列。其中，设定周期可以是预先设置时间窗口，例如2小时，根据当前的系统时间判断向压缩序列中写入数据的操作的执行时间是否达到2小时，若是，则将压缩序列中的压缩数据导出，形成数据块，将该数据块写入设定的压缩数据。清空压缩序列后，将接收到的第一个时序数据写入压缩序列的首个元素的位置，并在清空压缩序列后，启用一个新的压缩数据，用于存储当前压缩序列中的压缩数据。可选的，上述压缩数据可以与压缩序列设置在一起，还可以在将压缩数据存入压缩数组后，将压缩数组存储于设定存储空间，保存压缩序列与设定存储空间的标识信息的关联信息。这样设置的好

处在于通过时间窗口的方式将待写入的时序数据分成多个与时间相关的压缩数据块,在读取时序数据时,查询压缩序列或各个压缩数据块,确定目标压缩数据,提高了查询效率,同时对目标压缩数据进行解压而不是解压所有压缩数据,提高了读取效率。

[0056] 在上述技术方案的基础上,若检测到数据备份事件被触发,分别扫描每个缓存块中跳表中的时序数据项,获取时序数据项对应的压缩数据;对该压缩数据进行序列化处理,并将序列化处理后的压缩数据写入每个缓存块对应的备份文件,将该备份文件存储于设定磁盘。可选的,可以将备份文件存储于设定磁盘的数据目录下。其中,缓存系统检测到退出信号触发数据备份事件。可以采用PB(Protocol Buffer,协议缓存)技术对压缩数据进行序列化处理。Protocol Buffer是一种数据描述语言,能够将结构化数据序列化,可以使用该技术来持久化数据或者将数据序列化成网络传输格式的数据。这样设置的好处在于,待将所有缓存块内中的压缩数据备份至磁盘后,缓存系统正常退出(如关机),避免发生因缓存系统退出导出缓存于内存中的数据丢失的问题。

[0057] 在上述技术方案的基础上,检测到数据加载事件被触发,获取设定磁盘的设定目录下的备份文件;对该备份文件进行解序列化处理,得到压缩数据块;采用设定解压缩算法对上述压缩数据块进行解压缩处理,得到解压缩处理后的时序数据,将所述时序数据写回内存。需要说明的是,通过后台线程执行将备份文件写回内存操作,并对缓存块中正常的时序数据写入造成影响。设定解压缩算法在本发明实施例中有具体说明,此处不再赘述。这样设置的好处在于,在缓存系统开机后,由磁盘将备份文件写会到内存,避免因缓存系统关机而丢失的问题发生。

[0058] 图5是本发明实施例提供的另一种时序数据的存储方法的流程图,该方法包括:

[0059] 步骤510、获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识。

[0060] 步骤520、根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置。

[0061] 步骤530、获取所述待存储的时序数据中的时间戳和时序数据项的值。

[0062] 示例性的,获取待存储的时序数据中的时间戳 t_i 和时序数据项的值, i 是正整数,表示待存储的时序数据在时序数据流中的顺序。

[0063] 步骤540、计算所述时间戳与压缩序列中首个时序数据的时间戳之间的第一差值,计算所述第一差值与所述压缩序列中已写入的最后一条压缩数据的时间戳压缩值之间的第二差值。

[0064] 示例性的,由压缩序列中读取首个时序数据的时间戳 t_1 ,计算时间戳 t_i 与首个时序数据的时间戳 t_1 之间的第一差值 Δt_1 。获取压缩序列中已写入的最后一条压缩数据的时间戳压缩值 t'_{i-1} ,计算第一差值 Δt_1 与时间戳压缩值 t'_{i-1} 之间的第二差值 Δt_2 。

[0065] 步骤550、计算所述时序数据项的值与首个时序数据中的时序数据项的值的异或运算结果,根据所述异或运算结果匹配预设压缩策略。

[0066] 需要说明的是,该预设压缩策略包括:若异或运算结果为0,则确定所述待存储的时序数据中的时序数据项的值的压缩值为0;若异或运算结果为非0,则根据所述异或运算结果中的每个非0值以及与所述非0值相邻的0的数量确定所述待存储的时序数据中的时序数据项的值的压缩值。

[0067] 示例性的,对待存储的时序数据中的时序数据项的值与首个时序数据中的时序数据项的值进行异或运算,得到异或运算结果,基于异或运算结果匹配预设压缩策略。

[0068] 步骤560、基于所述预设压缩策略对所述待存储的时序数据中的时序数据项的值进行无损压缩,将无损压缩结果与所述第二差值的组合作为压缩数据。

[0069] 示例性的,判断异或运算结果是否为0;若是,则确定该待存储的时序数据中的时序数据项的值的压缩值为0;否则,记录异或运算结果中的每个非0值以及与该非0值相邻的0的数量,根据每个非0值和该非0值前后的0值的数量待存储的时序数据中的时序数据项的值,得到待存储的时序数据中的时序数据项的值的压缩值(即无损压缩结果),例如,异或运算结果是001000,压缩值就是213,采用类似的形式压缩时序数据项的值。将上述第二差值与该待存储的时序数据中的时序数据项的值的压缩值的组合作为压缩数据。需要说明的是,上述示例是对压缩算法的一个实现形式的说明,并不代表真实情况下的压缩数据的形式。

[0070] 步骤570、将所述压缩数据存储于所述存储位置上的压缩序列中。

[0071] 步骤580、按照设定周期将所述压缩序列中的所述压缩数据导出至设定的压缩数组,以清空所述压缩序列。

[0072] 本发明实施例的技术方案,通过计算待存储的时序数据项的值与首个时序数据中的时序数据项的值的异或运算结果的方式对时序数据项的值进行压缩,便于在读取过程中,采用对时序数据项的值的压缩值与首个时序数据中的时序数据项的值进行异或运算的方式,对压缩数据中时序数据项的值的压缩值进行还原,实现通过简单运算完成数据解压的效果。

[0073] 图6是本发明实施例提供的另一种时序数据的存储方法的流程图,该方法包括:

[0074] 步骤610、获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识。

[0075] 步骤620、根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置。

[0076] 步骤630、获取所述待存储的时序数据中的时间戳和时序数据项的值。

[0077] 步骤640、获取所述压缩序列对应的开始时间戳和结尾时间戳。

[0078] 需要说明的是,开始时间戳为该压缩序列内已存储的时序数据中最小的时间戳,该结尾时间戳为所述压缩序列内已存储的时序数据中最大的时间戳。在向时序序列中写入第一个时序数据中的时间戳后,将该时间戳作为开始时间戳。待接收到下一相邻的待存储时序数据的时间戳时,将新接收的时间戳与开始时间戳进行比较。若新接收的时间戳大于开始时间戳,则将新接收的时间戳作为结尾时间戳;若新接收的时间戳小于开始时间戳,则将原来的开始时间戳作为结尾时间戳,将新接收的时间戳作为开始时间戳。

[0079] 示例性的,获取压缩序列的属性信息中包含的开始时间戳和结尾时间戳。

[0080] 步骤650、将所述待存储的时序数据中的时间戳与所述开始时间戳和所述结尾时间戳进行比较,根据比较结果更新所述压缩序列对应的开始时间戳或结尾时间戳。

[0081] 示例性的,将待存储的时序数据中的时间戳与所获取的开始时间戳和结尾时间戳进行比较;若该时间戳小于开始时间戳,则将该时间戳作为新的开始时间戳;若该时间戳大于结尾时间戳,则将该时间戳作为新的结尾时间戳;基于上述比较结果更新压缩序列的属

性信息中包含的开始时间戳和结尾时间戳。

[0082] 步骤660、根据所述时间戳、首个时序数据中的时间戳与压缩序列中已写入的最后一个压缩数据的时间戳压缩值之间的差值、以及所述时序数据项的值与首个时序数据中的时序数据项的值之间的相似度,对所述待存储的时序数据中的时间戳和时序数据项的值进行无损压缩处理,得到压缩数据。

[0083] 步骤670、将所述压缩数据存储于所述存储位置上的压缩序列中。

[0084] 步骤680、按照设定周期将所述压缩序列中的所述压缩数据导出至设定的压缩数组,以清空所述压缩序列。

[0085] 步骤690、将所述压缩序列的所述开始时间戳和结尾时间戳存储至所述压缩数组。

[0086] 本发明实施例的技术方案,通过为压缩序列或压缩数组添加开始时间戳和结尾时间戳的属性信息,实现对压缩序列已经写入的时序数据的时间戳的范围进行记录,为时序数据读取过程中快速查找到存储待读取的时序数据的压缩序列或压缩数组提供便利,可以有效地提高读取效率。

[0087] 图7是本发明实施例提供的一种时序数据的读取方法的流程图,该方法可以由时序数据的读取装置来执行,其中,该装置可以由软件和/或硬件实现,一般可集成在电子设备中。如图7所示,该方法包括:

[0088] 步骤710、获取时序数据项和目标时间的的时间戳,基于所述时序数据项确定待读取的时序数据的存储位置标识。

[0089] 需要说明的是,目标时间的的时间戳是待读取的时序数据的时间戳范围。获取时序数据项和目标时间的的时间戳可以确定要读取哪些时序数据项在哪段时间内的时序数据。

[0090] 示例性的,在获取待读取的时序数据的时序数据项之后,根据哈希函数计算该时序数据项对应的哈希值;根据缓存块集合中的缓存块的数量确定缓存节点数量,对该哈希值和缓存节点数量进行取模运算,得到缓存节点标识,其中,所述缓存节点数量与所述缓存块的数量存在对数关系;基于该缓存节点标识与缓存块的对应关系,确定存储待读取的时序数据的缓存块,将该缓存块的标识信息作为存储位置标识。上述基于时序数据项确定存储位置标识的实现过程,与本发明实施例中时序数据存储过程中基于时序数据项确定存储位置标识的方法类似,此处不再赘述。

[0091] 步骤720、根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置。

[0092] 示例性的,根据存储位置标识(也就是缓存块的标识信息)查询缓存块集合可以确定与存储位置标识对应的目标缓存块。目标缓存块中存储有时序数据项与存储位置的关联关系,基于待读取的时序数据中的时序数据项以及上述关联关系,可以确定待读取的时序数据的存储位置。

[0093] 步骤730、根据所述目标时间的的时间戳由所述存储位置读取压缩数据。

[0094] 本发明实施例中,获取存储位置上的压缩序列的属性信息,基于该属性信息获取该压缩序列对应的开始时间戳和结尾时间戳,其中,开始时间戳为所述压缩序列内已存储的时序数据中最小的时间戳,结尾时间戳为所述压缩序列内已存储的时序数据中最大的时间戳。若该目标时间的的时间戳属于由该开始时间戳和结尾时间戳构成的时间戳区间,则由该压缩序列中获取已存储的压缩数据。若该目标时间的的时间戳不属于所述时间戳区间,则

根据该目标时间的时间戳匹配设定压缩数组内的开始时间戳和结尾时间戳,获取匹配成功的设定压缩数组内的压缩数据。

[0095] 示例性的,获取存储位置上的压缩序列对应的开始时间戳和结尾时间戳。将目标时间的时间戳与上述压缩序列对应的开始时间戳和结尾时间戳进行比对。若该目标时间的时间戳属于上述由开始时间戳和结尾时间戳构成的时间戳区间,则由该压缩序列中获取已存储的压缩数据。若该目标时间的时间戳不属于上述由开始时间戳和结尾时间戳构成的时间戳区间,则根据目标时间的时间戳匹配设定压缩数组内的开始时间戳和结尾时间戳,获取匹配成功的设定压缩数组内的压缩数据。其中,匹配成功的含义是目标时间的时间戳全部或部分属于压缩数组的开始时间戳和结尾时间戳构成的时间戳区间。

[0096] 需要说明的是,若待读取的时序数据的目标时间的时间戳全部属于上述由开始时间戳和结尾时间戳构成的时间戳区间,则确定待读取的时序数据全部位于压缩序列中,读取该压缩序列中已存储的全部压缩数据。若待读取的时序数据的目标时间的时间戳全部不属于上述由开始时间戳和结尾时间戳构成的时间戳区间,则根据目标时间的时间戳查询设定压缩数组。一些示例中该目标时间的时间戳属于一个压缩数组,则读取该压缩数组的全部压缩数据。另一些示例中,该目标时间的时间戳属于至少两个压缩数组,则分别读取每个压缩数组的全部压缩数据。若读取的时序数据的目标时间的时间戳部分属于压缩序列部分属于压缩数组,则分别读取该压缩序列中已存储的全部压缩数据,以及压缩数组中的全部压缩数据。

[0097] 步骤740、采用设定解压缩算法对所述压缩数据进行解压得到待读取的时序数据。

[0098] 示例性的,获取该压缩序列或匹配成功的设定压缩数组中存储的首个时序数据的时间戳和时序数据项的值。需要说明的是,若压缩数据全部由压缩序列中读取,则可以获取压缩序列中存储的首个时序数据的时间戳和时序数据项的值。若压缩数据全部由匹配成功的压缩数组读取,则可以获取匹配成功的压缩数组中存储的首个时序数据的时间戳和时序数据项的值。若既由压缩序列中读取压缩数据又由匹配成功的压缩数组中读取压缩数据,则分别获取序列中以及匹配成功的压缩数组中存储的首个时序数据的时间戳和时序数据项的值。

[0099] 获取该压缩数据中的待解压的时序数据的时间戳压缩值,计算该时间戳压缩值、首个时序数据的时间戳与待解压的时序数据的上一时间相邻的时序数据的时间戳压缩值的和值,得到该待解压的时序数据中的时间戳。需要说明的是,若压缩数据全部由压缩序列中读取,则计算待解压的时序数据的时间戳压缩值、所获取的压缩序列中存储的首个时序数据的时间戳与待解压的时序数据的上一时间相邻的时序数据的时间戳压缩值的和值,得到该待解压的时序数据中的时间戳。若压缩数据全部由匹配成功的压缩数组读取,则计算待解压的时序数据的时间戳压缩值、待解压的时序数据对应的压缩数组的首个时序数据的时间戳与待解压的时序数据的上一时间相邻的时序数据的时间戳压缩值的和值,得到该待解压的时序数据中的时间戳。例如,待解压的时序数据 a_i 属于压缩数组A,则计算解压的时序数据 a_i 的时间戳压缩值 t'_i 、压缩数组A的首个时序数据的时间戳 t_1 以及解压的时序数据 a_i 的上一时间相邻的时序数据的时间戳压缩值 t'_{i-1} 的和值,得到该待解压的时序数据 a_i 中的时间戳 t_i 。待解压的时序数据 b_i 属于压缩数组B,则计算解压的时序数据 b_i 的时间戳压缩值 T'_i 、压缩数组B的首个时序数据的时间戳 T_1 以及解压的时序数据 b_i 的上一时间相邻的时

序数据的时间戳压缩值 T'_{i-1} 的和值,得到该待解压的时序数据 b_i 中的时间戳 T_i 。若既由压缩序列中读取压缩数据又由匹配成功的压缩数组中读取压缩数据,则根据压缩序列中存储的首个时序数据的时间戳对由压缩序列读取的压缩数据的时间戳压缩值进行解压,根据压缩数组中首个时序数据的时间戳对由压缩数组读取的压缩数据的时间戳压缩值进行解压。

[0100] 获取压缩数据中的待解压的时序数据的时序数据项的值的压缩值,计算压缩值与首个时序数据中的时序数据项的值的异或运算结果,得到待解压的时序数据中的时序数据项的值。需要说明的是,若压缩数据全部由压缩序列中读取,则可以计算待解压的时序数据的时序数据项的值的压缩值与由压缩序列中获取的首个时序数据的时序数据项的值的异或运算结果,得到待解压的时序数据中的时序数据项的值。若压缩数据全部由匹配成功的压缩数组读取,则可以计算待解压的时序数据的时序数据项的值的压缩值与由对应压缩数组中存储的首个时序数据的时间戳和时序数据项的值的异或运算结果,得到待解压的时序数据中的时序数据项的值。若既由压缩序列中读取压缩数据又由匹配成功的压缩数组中读取压缩数据,则根据压缩数据来源分别计算待解压的时序数据的时序数据项的值的压缩值与由压缩序列中获取的首个时序数据的时序数据项的值的异或运算结果、以及待解压的时序数据的时序数据项的值的压缩值与由对应压缩数组中存储的首个时序数据的时间戳和时序数据项的值的异或运算结果,得到待解压的时序数据中的时序数据项的值。

[0101] 将目标时间的时间戳与解压后的时序数据中的时间戳进行匹配,根据匹配结果由解压后的时序数据中确定待读取的时序数据。需要说明的是,所获取的压缩数据可能多于待读取的压缩数据,所以,在对压缩数据进行解压后,可以基于目标时间的时间戳与解压后的时序数据中的时间戳进行匹配,获取时间戳属于目标时间的时间戳范围的解压后的时序数据,作为待读取的时序数据。

[0102] 本发明实施例的技术方案,通过首先将目标时间的时间戳与压缩序列的开始时间戳和结尾时间戳比较,若目标时间的时间戳不完全属于压缩序列的开始时间戳和结尾时间戳构成的时间戳区间,再根据目标时间的时间戳查询压缩数组,可以快速搜索到包含有待读取的时序数据的压缩数据,再通过设定解压算法对压缩数据进行解压,采用目标时间的时间戳定位到所有解压后的时序数据中的待读取时序数据。采用本发明实施例的技术方案,可以由与时间关联的压缩数据中快速查询到待读取的时序数据,可以避免发生因多次执行查询指令导致的查询延迟问题,降低了时序数据的查询延迟,提高了读取效率。

[0103] 图8是本发明实施例提供的一种时序数据的存储装置的结构框图,该装置可由软件和/或硬件实现,一般集成在电子设备中,可通过执行时序数据的存储方法来进行时序数据存储。如图8所示,该装置包括:

[0104] 存储位置标识确定模块810,用于获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识,其中,所述时序数据包含所述时序数据项、时序数据项的值和时间戳;

[0105] 存储位置确定模块820,用于根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

[0106] 数据存储模块830,用于采用设定压缩算法对所述时序数据项的值和所述时间戳进行压缩处理,并将压缩数据存储于所述存储位置。

[0107] 本发明实施例中提供的时序数据的存储装置,通过采用设定压缩算法对待存储的

时序数据包含的时序数据项的值和时间戳进行压缩的方式,得到与时间关联的压缩数据,并将该压缩数据存储于时序数据项对应的目标缓存块中,克服了时序数据大量占用设备的存储空间缺陷,提高了空间利用率;此外,由于缓存的是与时间关联的压缩数据,在对时序数据进行查询时,可以通过执行一次查询指令获得一段时间内的多条时序数据记录,避免发生因多次执行查询指令导致的查询延迟问题,降低了查询延时。

[0108] 可选的,存储位置标识确定模块810具体用于:

[0109] 获取待存储的时序数据的时序数据项,根据哈希函数计算所述时序数据项对应的哈希值;

[0110] 根据缓存块集合中的缓存块的数量确定缓存节点数量,对所述哈希值和缓存节点数量进行取模运算,得到缓存节点标识,其中,所述缓存节点数量与所述缓存块的数量存在对数关系;

[0111] 基于所述缓存节点标识与缓存块的对应关系,确定用于存储所述时序数据的缓存块,将所述缓存块的标识信息作为存储位置标识。

[0112] 可选的,数据存储模块830包括:

[0113] 数据获取子模块,用于获取所述待存储的时序数据中的时间戳和时序数据项的值;

[0114] 压缩子模块,用于根据所述时间戳、首个时序数据中的时间戳与压缩序列中已写入的最后一个压缩数据的时间戳压缩值之间的差值、以及所述时序数据项的值与首个时序数据中的时序数据项的值之间的相似度,对所述待存储的时序数据中的时间戳和时序数据项的值进行无损压缩处理,得到压缩数据;

[0115] 数据存储子模块,用于将所述压缩数据存储于所述存储位置上的压缩序列中;其中,所述首个时序数据是指所述压缩序列最近一次被清空后所述压缩序列中被存储的第一个时序数据。

[0116] 可选的,压缩子模块具体用于:

[0117] 计算所述时间戳与压缩序列中首个时序数据的时间戳之间的第一差值,计算所述第一差值与所述压缩序列中已写入的最后一条压缩数据的时间戳压缩值之间的第二差值;

[0118] 计算所述时序数据项的值与首个时序数据中的时序数据项的值的异或运算结果,根据所述异或运算结果匹配预设压缩策略;

[0119] 基于所述预设压缩策略对所述待存储的时序数据中的时序数据项的值进行无损压缩,将无损压缩结果与所述第二差值的组合作为压缩数据。

[0120] 可选的,所述预设压缩策略包括:

[0121] 若异或运算结果为0,则确定所述待存储的时序数据中的时序数据项的值的压缩值为0;

[0122] 若异或运算结果为非0,则根据所述异或运算结果中的每个非0值以及与所述非0值相邻的0的数量确定所述待存储的时序数据中的时序数据项的值的压缩值。

[0123] 可选的,还包括:

[0124] 时间戳获取模块,用于在获取所述待存储的时序数据中的时间戳之后,获取所述压缩序列对应的开始时间戳和结尾时间戳,其中,所述开始时间戳为所述压缩序列内已存储的时序数据中最小的时间戳,所述结尾时间戳为所述压缩序列内已存储的时序数据中最

大的时间戳；

[0125] 时间戳更新模块,用于将所述待存储的时序数据中的时间戳与所述开始时间戳和所述结尾时间戳进行比较,根据比较结果更新所述压缩序列对应的开始时间戳或结尾时间戳。

[0126] 可选的,还包括:

[0127] 数据导出模块,用于按照设定周期将所述压缩序列中的所述压缩数据导出至设定的压缩数组,以清空所述压缩序列;

[0128] 时间戳存储模块,用于将所述压缩序列的所述开始时间戳和结尾时间戳存储至所述压缩数组。

[0129] 图9是本发明实施例提供的一种时序数据的读取装置的结构框图,该装置可由软件和/或硬件实现,一般集成在电子设备中,可通过执行时序数据的读取方法来读取时序数据。如图9所示,该装置包括:

[0130] 读取位置标识确定模块910,用于获取时序数据项和目标时间,基于所述时序数据项确定待读取的时序数据的读取位置标识;

[0131] 读取位置确定模块920,用于根据所述读取位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的读取位置;

[0132] 压缩数据读取模块930,用于根据所述目标时间的的时间戳由所述存储位置读取压缩数据;

[0133] 数据解压模块940,用于采用设定解压缩算法对所述压缩数据进行解压得到待读取的时序数据。

[0134] 本发明实施例中提供的时序数据的读取装置,通过首先将目标时间的的时间戳与压缩序列的开始时间戳和结尾时间戳比较,若目标时间的的时间戳不完全属于压缩序列的开始时间戳和结尾时间戳构成的时间戳区间,再根据目标时间的的时间戳查询压缩数组,可以快速搜索到包含有待读取的时序数据的压缩数据,再通过设定解压算法对压缩数据进行解压,采用目标时间的的时间戳定位到所有解压后的时序数据中的待读取时序数据。采用本发明实施例的技术方案,可以由与时间关联的压缩数据中快速查询到待读取的时序数据,可以避免发生因多次执行查询指令导致的查询延迟问题,降低了时序数据的查询延迟,提高了读取效率。

[0135] 可选的,读取位置标识确定模块910具体用于:

[0136] 根据哈希函数计算所述时序数据项对应的哈希值;

[0137] 根据缓存块集合中的缓存块的数量确定缓存节点数量,对所述哈希值和缓存节点数量进行取模运算,得到缓存节点标识,其中,所述缓存节点数量与所述缓存块的数量存在对数关系;

[0138] 基于所述缓存节点标识与缓存块的对应关系,确定存储待读取的时序数据的缓存块,将所述缓存块的标识信息作为存储位置标识。

[0139] 可选的,压缩数据读取模块930具体用于:

[0140] 获取所述存储位置上的压缩序列对应的开始时间戳和结尾时间戳,其中,所述开始时间戳为所述压缩序列内已存储的时序数据中最小的时间戳,所述结尾时间戳为所述压缩序列内已存储的时序数据中最大的时间戳;

[0141] 若所述目标时间的时间戳属于由所述开始时间戳和结尾时间戳构成的时间戳区间,则由所述压缩序列中获取已存储的压缩数据;

[0142] 若所述目标时间的时间戳不属于所述时间戳区间,则根据所述目标时间的时间戳匹配设定压缩数组内的开始时间戳和结尾时间戳,获取匹配成功的设定压缩数组内的压缩数据。

[0143] 可选的,数据解压模块940具体用于:

[0144] 获取所述压缩序列或匹配成功的设定压缩数组中存储的首个时序数据的时间戳和时序数据项的值;

[0145] 获取所述压缩数据中的待解压的时序数据的时间戳压缩值,计算所述时间戳压缩值、首个时序数据的时间戳与待解压的时序数据的上一时间相邻的时序数据的时间戳压缩值的和值,得到所述待解压的时序数据中的时间戳;

[0146] 获取所述压缩数据中的待解压的时序数据的时序数据项的值的压缩值,计算所述压缩值与首个时序数据中的时序数据项的值的异或运算结果,得到所述待解压的时序数据中的时序数据项的值;

[0147] 将所述目标时间的时间戳与解压后的时序数据中的时间戳进行匹配,根据匹配结果由解压后的时序数据中确定待读取的时序数据。

[0148] 本发明实施例还提供了一种电子设备,该电子设备中可集成本发明实施例提供的时序数据的存储装置。图10是本发明实施例提供的一种电子设备的结构框图。电子设备可以包括存储器1010、处理器1020及存储在存储器1010上并可在处理器1020运行的计算机程序,该处理器1020执行该计算机程序时实现如本发明实施例所述的时序数据的存储方法。

[0149] 本发明实施例中提供的电子设备,通过采用设定压缩算法对待存储的时序数据包含的时序数据项的值和时间戳进行压缩的方式,得到与时间关联的压缩数据,并将该压缩数据存储于时序数据项对应的目标缓存块中,克服了时序数据大量占用设备的存储空间缺陷,提高了空间利用率;此外,由于缓存的是与时间关联的压缩数据,在对时序数据进行查询时,可以通过执行一次查询指令获得一段时间内的多条时序数据记录,避免发生因多次执行查询指令导致的查询延迟问题,降低了查询延迟。

[0150] 本发明实施例还提供了另一种电子设备,该电子设备中可集成本发明实施例提供的时序数据的读取装置。图11是本发明实施例提供的另一种电子设备的结构框图。电子设备可以包括存储器1110、处理器1120及存储在存储器1110上并可在处理器1120运行的计算机程序,该处理器1120执行该计算机程序时实现如本发明实施例所述的时序数据的读取方法。

[0151] 本发明实施例中提供的电子设备,通过首先将目标时间的时间戳与压缩序列的开始时间戳和结尾时间戳比较,若目标时间的时间戳不完全属于压缩序列的开始时间戳和结尾时间戳构成的时间戳区间,再根据目标时间的时间戳查询压缩数组,可以快速搜索到包含有待读取的时序数据的压缩数据,再通过设定解压算法对压缩数据进行解压,采用目标时间的时间戳定位到所有解压后的时序数据中的待读取时序数据。采用本发明实施例的技术方案,可以由与时间关联的压缩数据中快速查询到待读取的时序数据,可以避免发生因多次执行查询指令导致的查询延迟问题,降低了时序数据的查询延迟,提高了读取效率。

[0152] 本发明实施例还提供了一种包含计算机可执行指令的存储机制,所述计算机可执

行指令在由计算机处理器执行时用于执行时序数据的存储方法,该方法包括:

[0153] 获取待存储的时序数据,基于所述时序数据的时序数据项确定存储位置标识,其中,所述时序数据包含所述时序数据项、时序数据项的值和时间戳;

[0154] 根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

[0155] 采用设定压缩算法对所述时序数据项的值和所述时间戳进行压缩处理,并将压缩数据存储于所述存储位置。

[0156] 当然,本发明实施例所提供的一种包含计算机可执行指令的存储介质,其计算机可执行指令不限于如上所述的时序数据存储操作,还可以执行本发明任意实施例所提供的时序数据的存储方法中的相关操作。

[0157] 本发明实施例还提供了另一种包含计算机可执行指令的存储机制,所述计算机可执行指令在由计算机处理器执行时用于执行时序数据的读取方法,该方法包括:

[0158] 获取时序数据项和目标时间的时间戳,基于所述时序数据项确定待读取的时序数据的存储位置标识;

[0159] 根据所述存储位置标识由缓存块集合中确定目标缓存块,根据所述时序数据项由所述目标缓存块中确定对应的存储位置;

[0160] 根据所述目标时间的时间戳由所述存储位置读取压缩数据;

[0161] 采用设定解压缩算法对所述压缩数据进行解压得到待读取的时序数据。

[0162] 当然,本发明实施例所提供的一种包含计算机可执行指令的存储介质,其计算机可执行指令不限于如上所述的时序数据读取操作,还可以执行本发明任意实施例所提供的时序数据的读取方法中的相关操作。

[0163] 需要说明的是,存储介质——任何的各种类型的存储器设备或存储设备。术语“存储介质”旨在包括:安装介质,例如CD-ROM、软盘或磁带装置;计算机系统存储器或随机存取存储器,诸如DRAM、DDR RAM、SRAM、EDORAM,兰巴斯(Rambus) RAM等;非易失性存储器,诸如闪存、磁介质(例如硬盘或光存储);寄存器或其它相似类型的存储器元件等。存储介质可以还包括其它类型的存储器或其组合。另外,存储介质可以位于程序在其中被执行的第一计算机系统中,或者可以位于不同的第二计算机系统中,第二计算机系统通过网络(诸如因特网)连接到第一计算机系统。第二计算机系统可以提供程序指令给第一计算机用于执行。术语“存储介质”可以包括可以驻留在不同位置中(例如在通过网络连接的不同计算机系统中)的两个或更多存储介质。存储介质可以存储可由一个或多个处理器执行的程序指令(例如具体实现为计算机程序)。

[0164] 上述实施例中提供的时序数据的存储装置、存储介质、电子设备可执行本发明任意实施例所提供的时序数据的存储方法,具备执行该方法相应的功能模块和有益效果。未在上述实施例中详尽描述的技术细节,可参见本发明任意实施例所提供的时序数据的存储方法。

[0165] 上述实施例中提供的时序数据的读取装置、存储介质、电子设备可执行本发明任意实施例所提供的时序数据的读取方法,具备执行该方法相应的功能模块和有益效果。未在上述实施例中详尽描述的技术细节,可参见本发明任意实施例所提供的时序数据的读取方法。

[0166] 注意,上述仅为本发明的较佳实施例及所运用技术原理。本领域技术人员会理解,本发明不限于这里所述的特定实施例,对本领域技术人员来说能够进行各种明显的变化、重新调整和替代而不会脱离本发明的保护范围。因此,虽然通过以上实施例对本发明进行了较为详细的说明,但是本发明不仅仅限于以上实施例,在不脱离本发明构思的情况下,还可以包括更多其他等效实施例,而本发明的范围由所附的权利要求范围决定。

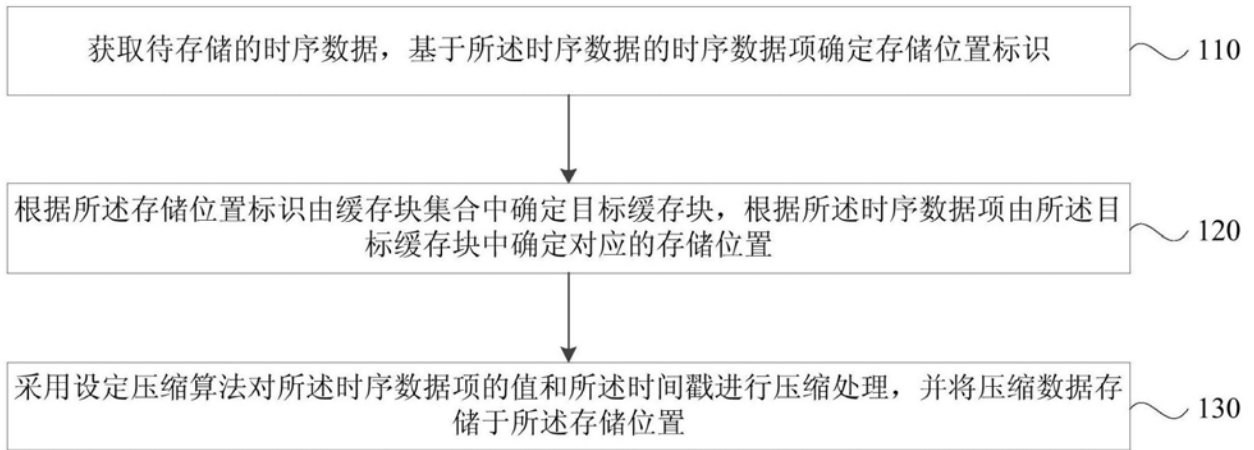


图1

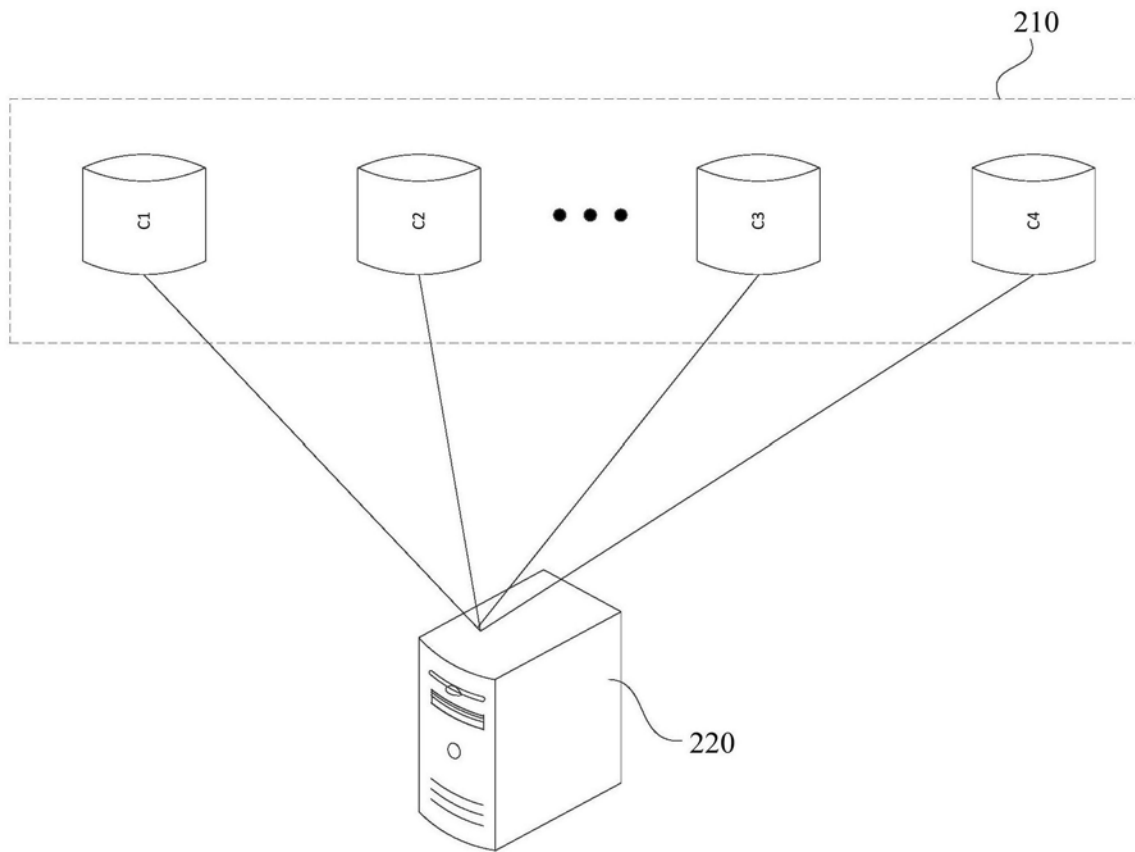


图2

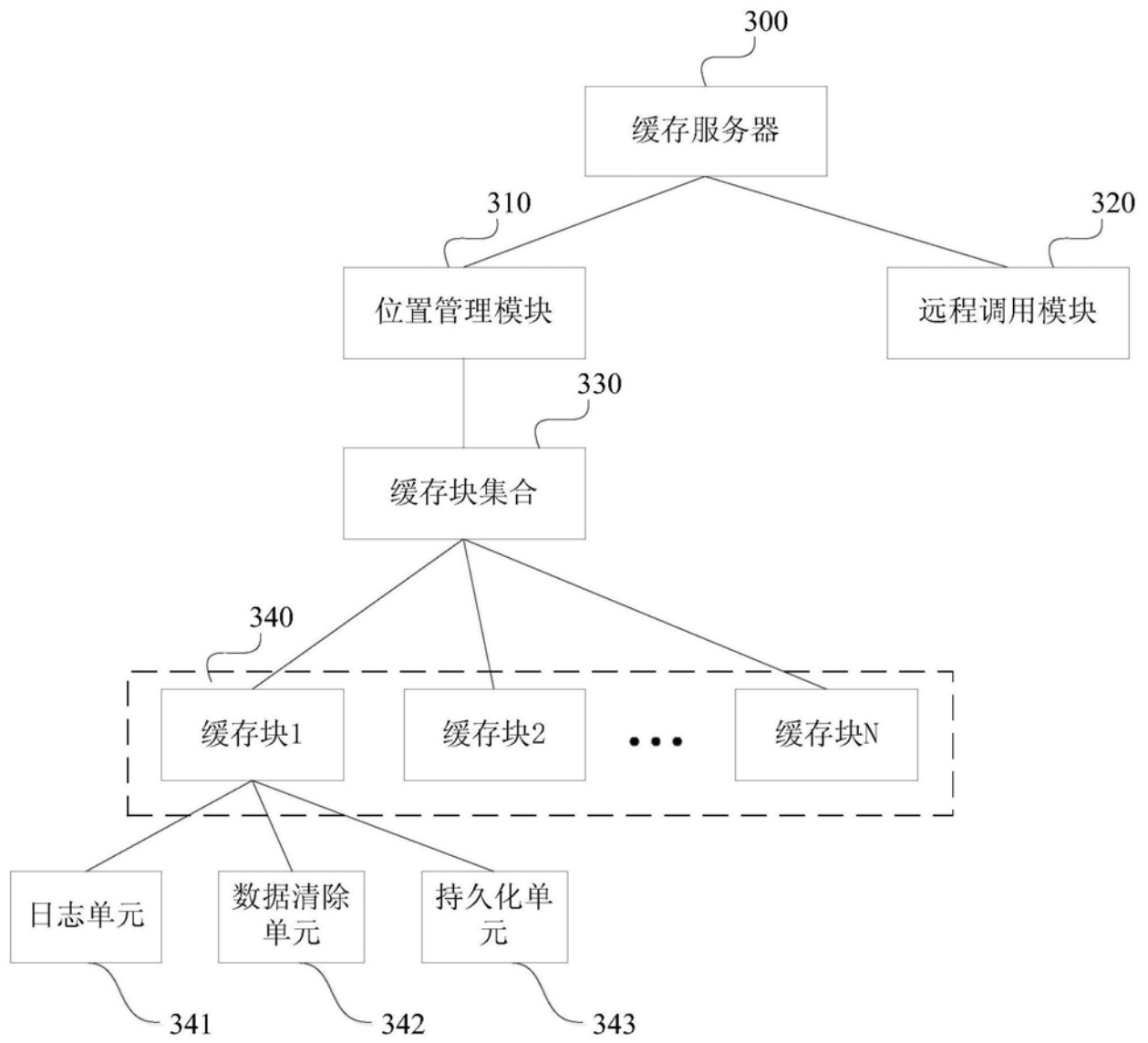


图3

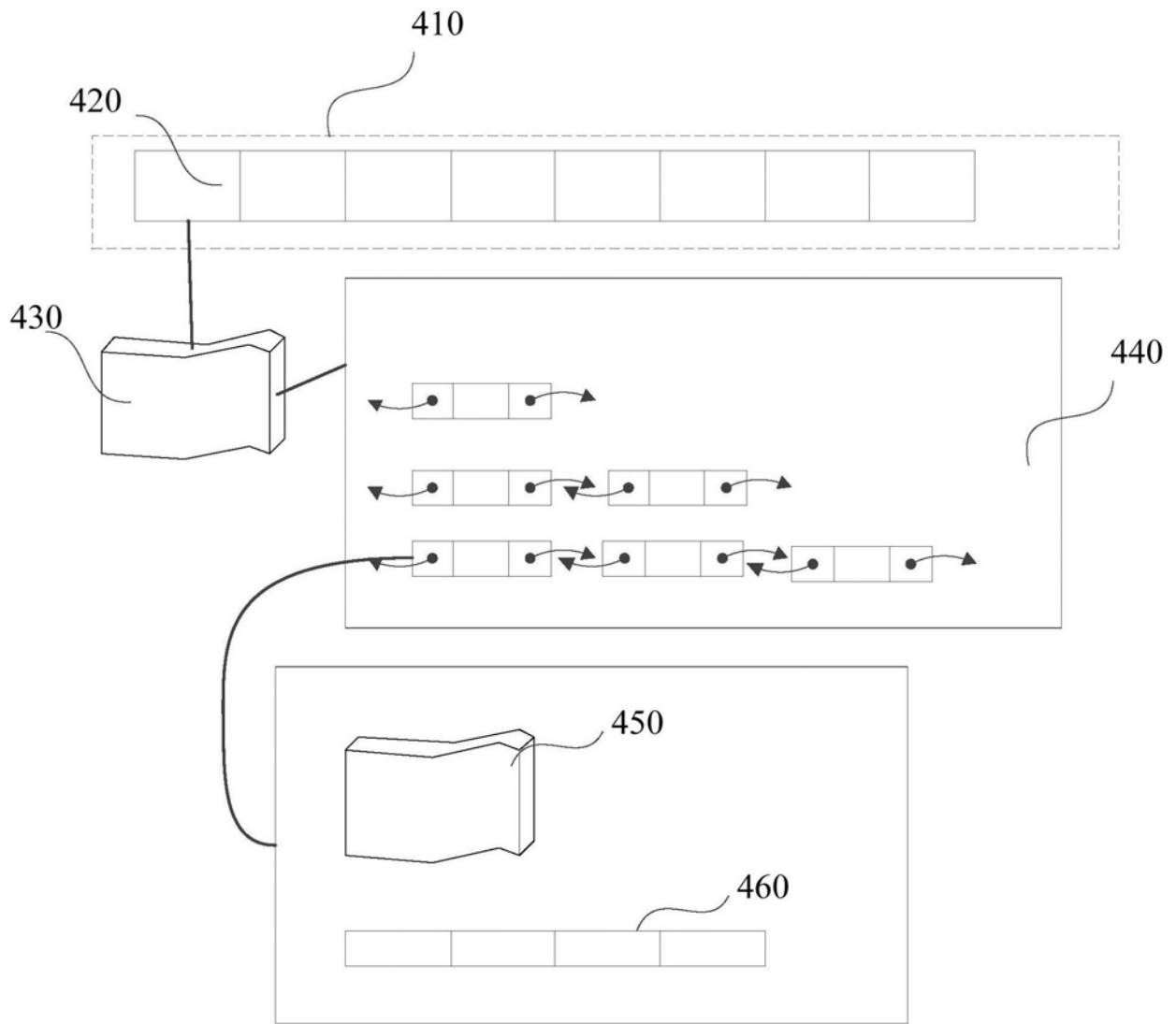


图4

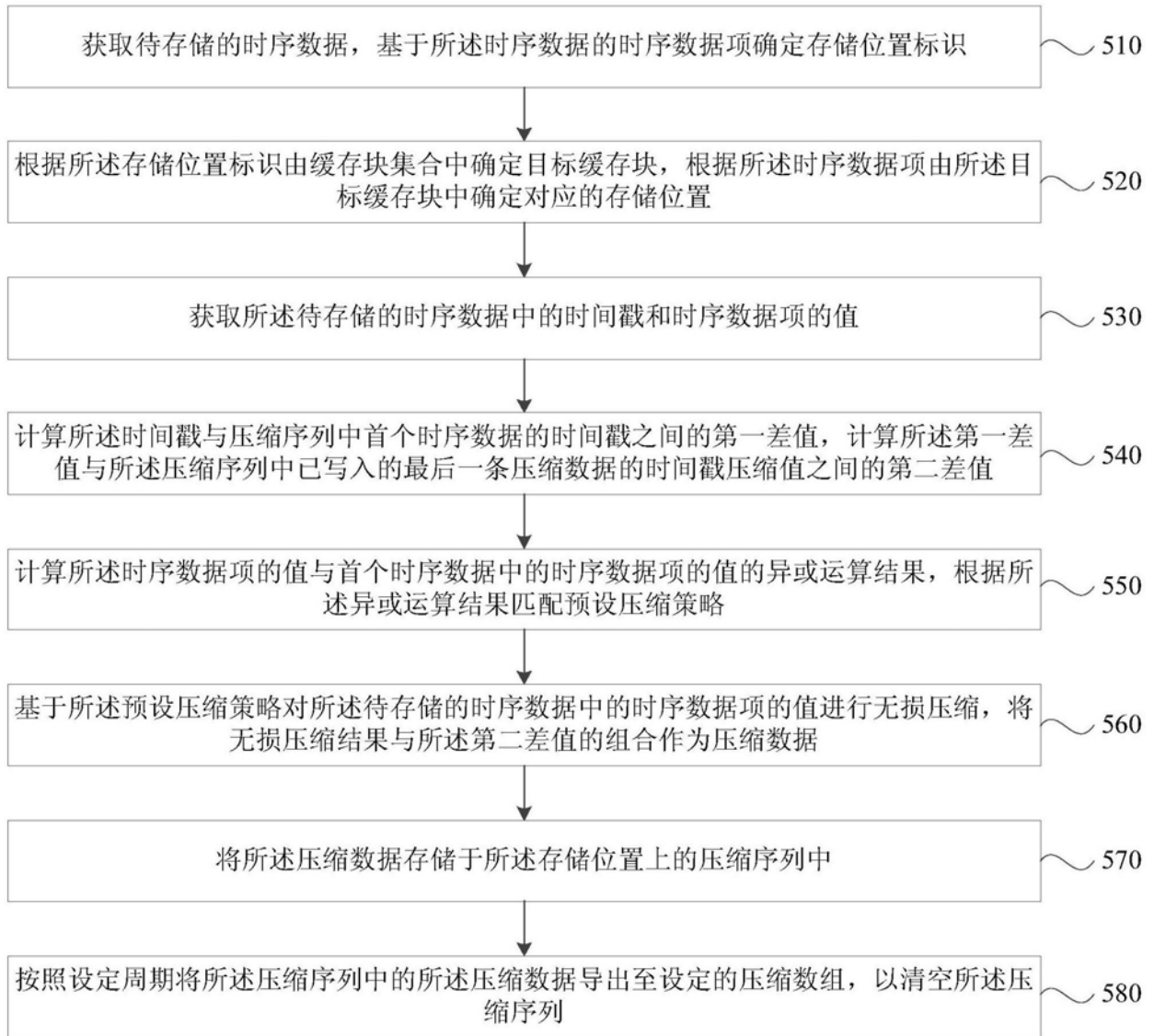


图5

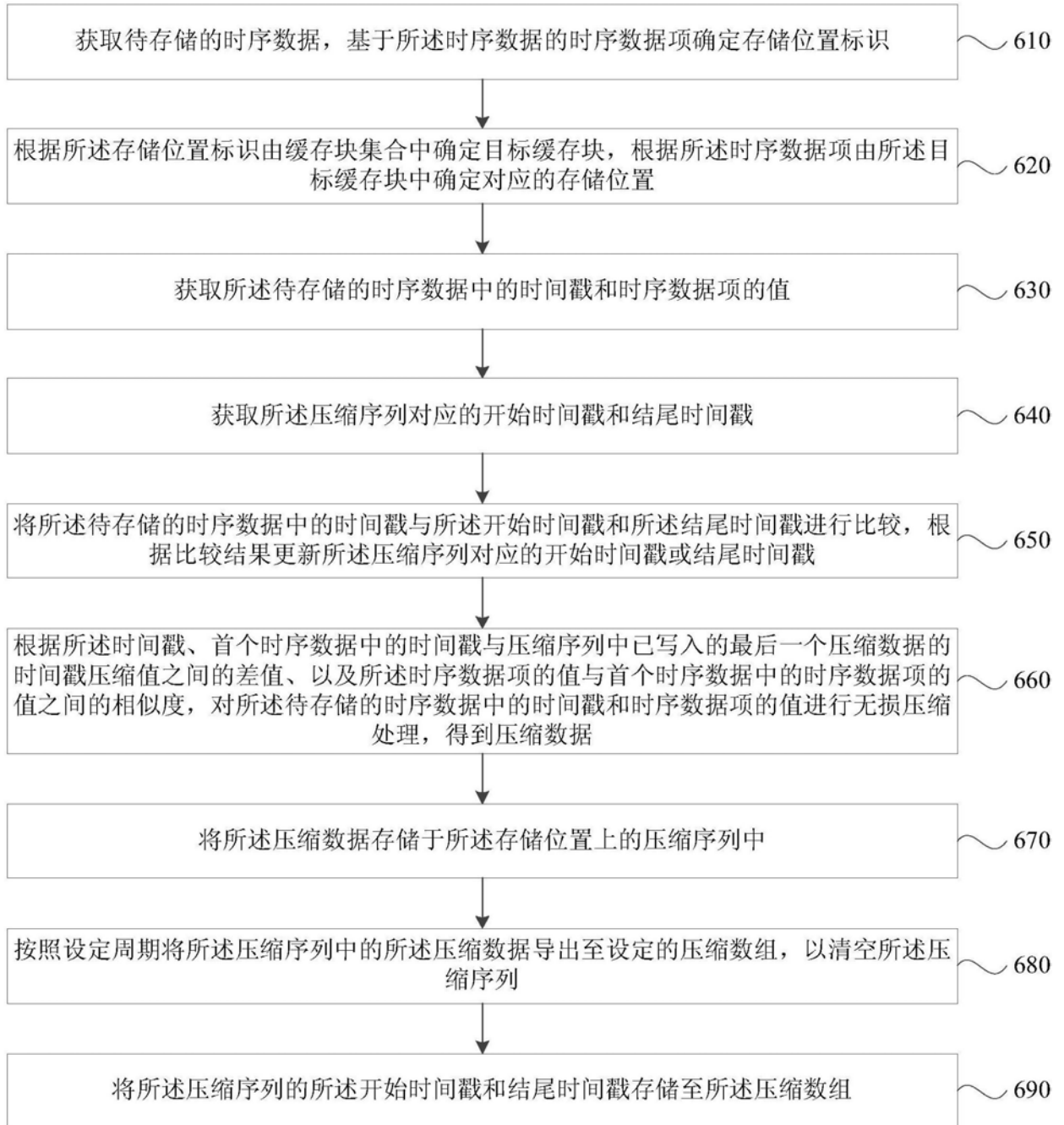


图6

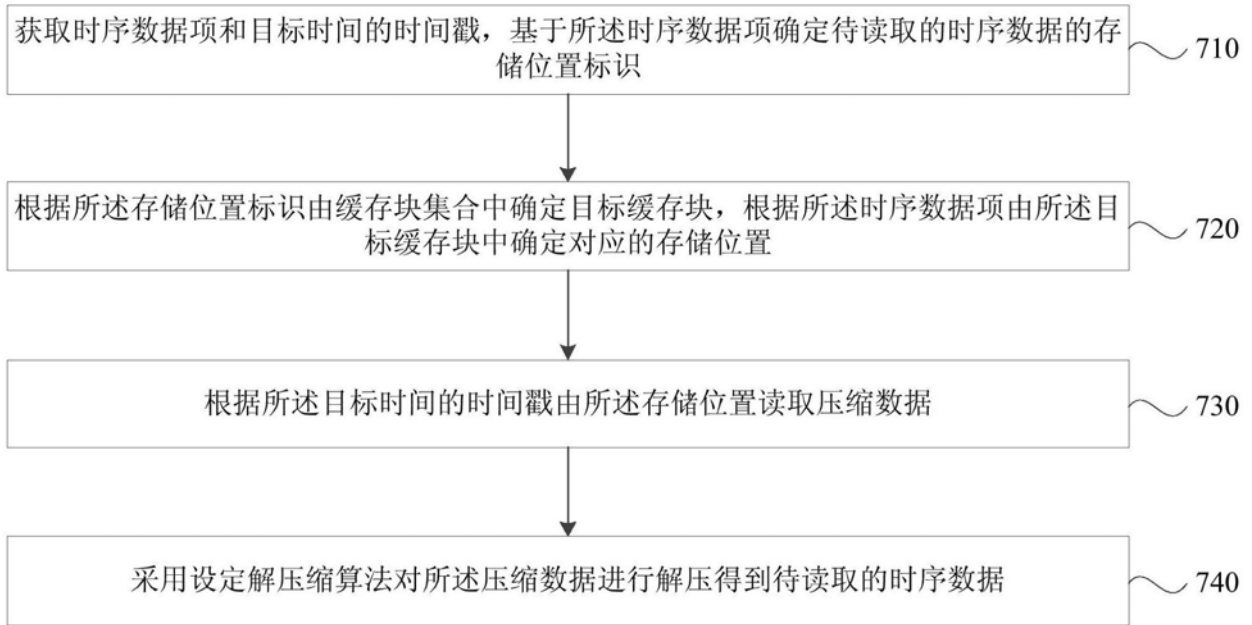


图7

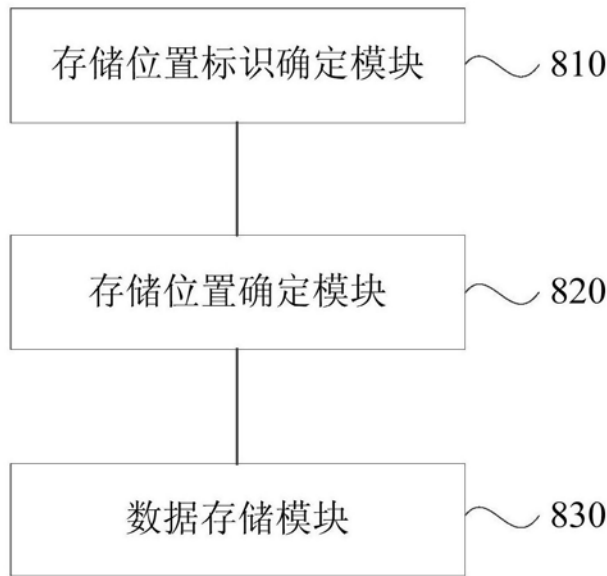


图8

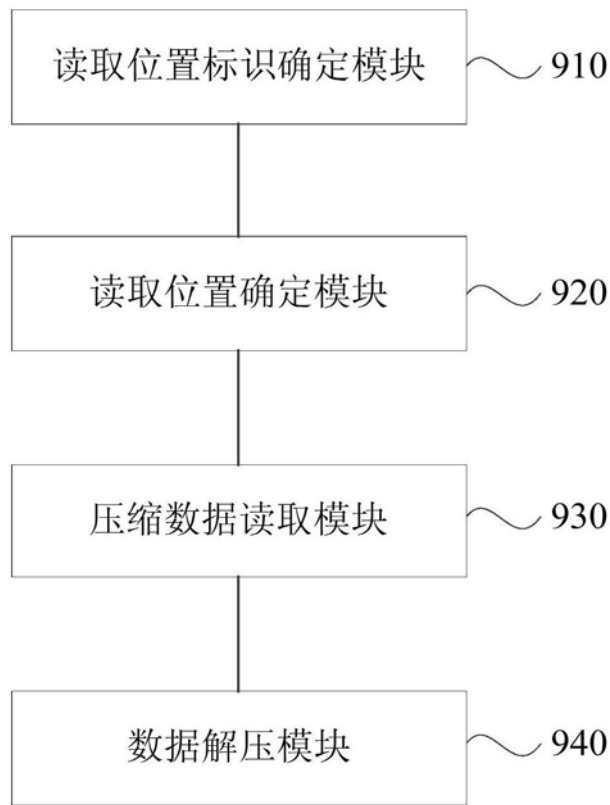


图9

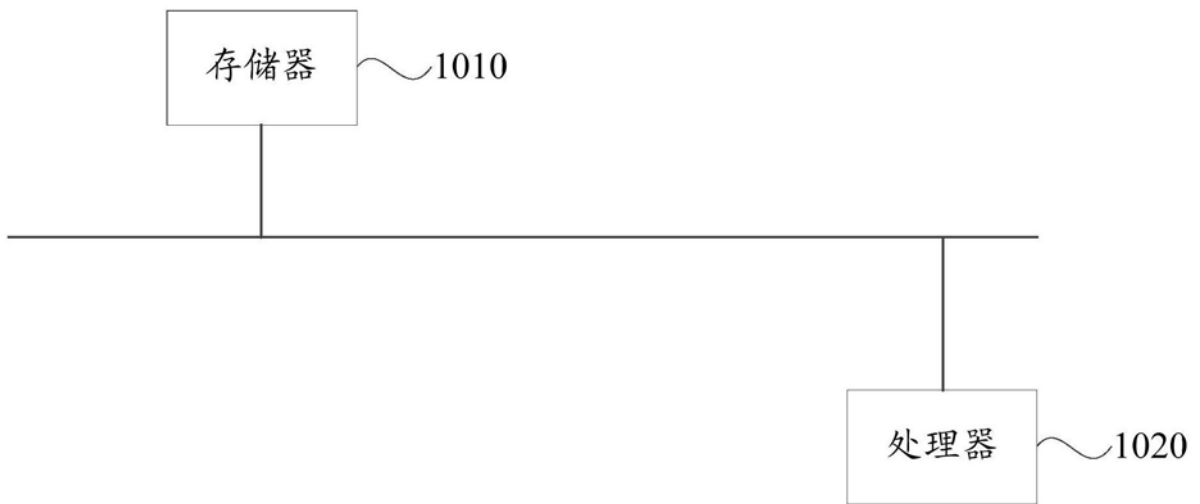


图10

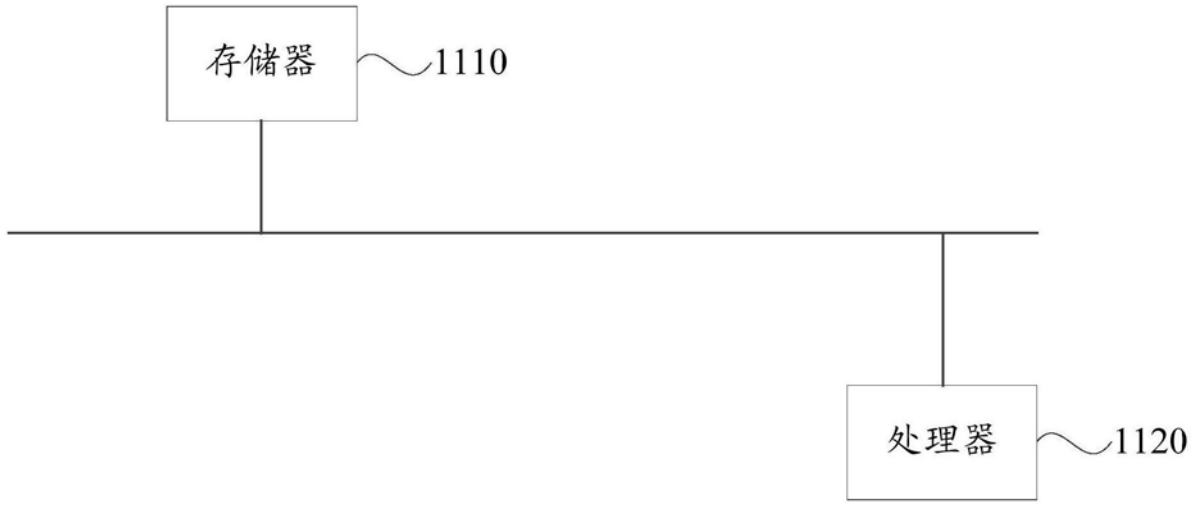


图11