



(12) 发明专利

(10) 授权公告号 CN 113918530 B

(45) 授权公告日 2022.05.13

(21) 申请号 202111519652.2

(22) 申请日 2021.12.14

(65) 同一申请的已公布的文献号
申请公布号 CN 113918530 A

(43) 申请公布日 2022.01.11

(73) 专利权人 北京达佳互联信息技术有限公司
地址 100085 北京市海淀区上地西路6号1
幢1层101D1-7

(72) 发明人 王军

(74) 专利代理机构 北京律智知识产权代理有限公司 11438
专利代理师 孙宝海 阚梓瑄

(51) Int. Cl.

G06F 16/176 (2019.01)

H04L 67/1097 (2022.01)

(56) 对比文件

CN 104391935 A, 2015.03.04

US 2016021026 A1, 2016.01.21

CN 110888858 A, 2020.03.17

CN 106202271 A, 2016.12.07

CN 109766324 A, 2019.05.17

审查员 穆滢

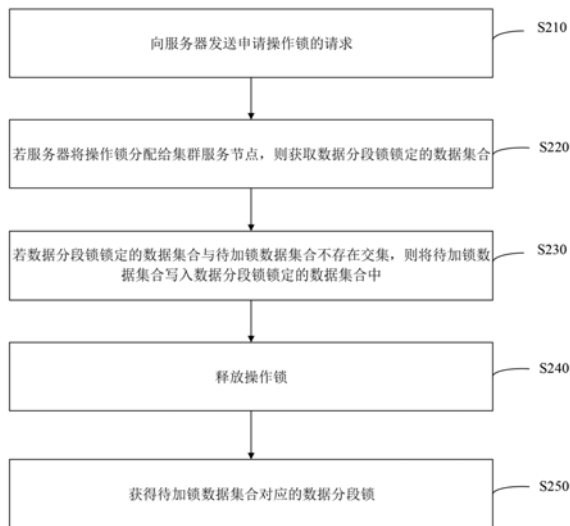
权利要求书3页 说明书12页 附图7页

(54) 发明名称

分布式锁的实现方法、装置、电子设备及介质

(57) 摘要

本公开关于一种分布式锁的实现方法、装置、电子设备及介质,涉及计算机技术领域。由集群服务节点执行的方法包括:向服务器发送申请操作锁的请求;若服务器将操作锁分配给集群服务节点,则获取数据分段锁锁定的数据集合;若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则将待加锁数据集合写入数据分段锁锁定的数据集合中;释放操作锁;获得待加锁数据集合对应的数据分段锁。该方法采用操作锁和数据分段锁的双重锁操作,实现多个服务节点对数据集合的不同分段加锁,解决相关技术中无法对大批量数据进行并发操作的问题。



1. 一种分布式锁的实现方法,应用于集群服务节点,其特征在于,包括:
向服务器发送申请操作锁的请求;
若所述服务器将操作锁分配给所述集群服务节点,则获取数据分段锁锁定的数据集合;
若所述数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则通过所述服务器生成所述待加锁数据集合对应的数据分段锁,将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中,以通过写入数据的方式锁定所述待加锁数据集合;
在将所述待加锁数据集合写入所述数据分段锁锁定的数据集合后,释放所述操作锁,其中所述操作锁负责锁定对数据分段的操作;
获得所述待加锁数据集合对应的数据分段锁;以及
执行业务操作,在业务操作完成后,将所述待加锁数据集合从所述数据分段锁锁定的数据集合中移除,释放所述待加锁数据集合对应的数据分段锁。
2. 根据权利要求1所述的方法,其特征在于,将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中,包括:
确定所述待加锁数据集合对应的数据分段锁;
根据所述待加锁数据集合对应的数据分段锁的唯一标识,将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中。
3. 根据权利要求1所述的方法,其特征在于,所述方法还包括:
若所述数据分段锁锁定的数据集合与所述待加锁数据集合存在交集,则进入等待状态;
等待锁定所述交集的数据分段锁被释放之后,申请数据分段锁。
4. 根据权利要求3所述的方法,其特征在于,所述方法还包括:
若在预设等待时间内所述锁定所述交集的数据分段锁未被释放,则释放所述操作锁。
5. 根据权利要求1所述的方法,其特征在于,所述方法还包括:
若所述服务器未将所述操作锁分配给所述集群服务节点,则进入等待状态;
等待持有所述操作锁的服务节点释放所述操作锁之后,申请所述操作锁。
6. 一种分布式锁的实现方法,应用于服务器,其特征在于,包括:
接收集群服务节点发送的申请操作锁的请求;
若操作锁未被其他服务节点持有,则将所述操作锁分配给所述集群服务节点;
若操作锁被其他服务节点持有,则令所述集群服务节点进入等待状态,等待持有所述操作锁的服务节点释放所述操作锁之后,将所述操作锁分配给所述集群服务节点;以及,
在将所述操作锁分配给所述集群服务节点之后,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,生成所述待加锁数据集合对应的数据分段锁,以通过所述集群服务节点将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中,然后所述集群服务节点释放所述操作锁,获得所述待加锁数据集合对应的数据分段锁,接着所述集群服务节点执行业务操作,在业务操作完成后,将所述待加锁数据集合从所述数据分段锁锁定的数据集合中移除,释放所述待加锁数据集合对应的数据分段锁,其中所述操作锁负责锁定对数据分段的操作。
7. 一种分布式锁的实现装置,应用于集群服务节点,其特征在于,包括:

申请锁模块,被配置为向服务器发送申请操作锁的请求;

数据获取模块,被配置为若所述服务器将操作锁分配给所述集群服务节点,则获取数据分段锁锁定的数据集合;

写入模块,被配置为若所述数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则通过所述服务器生成所述待加锁数据集合对应的数据分段锁,将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中,以通过写入数据的方式锁定所述待加锁数据集合;

释放锁模块,被配置为在将所述待加锁数据集合写入所述数据分段锁锁定的数据集合后,释放所述操作锁,其中所述操作锁负责锁定对数据分段的操作;

锁获取模块,被配置为获得所述待加锁数据集合对应的数据分段锁;

执行模块,被配置为执行业务操作;以及,所述释放锁模块被配置为:在业务操作完成后,将所述待加锁数据集合从所述数据分段锁锁定的数据集合中移除,释放所述待加锁数据集合对应的数据分段锁。

8. 根据权利要求7所述的装置,其特征在于,所述写入模块被配置为:

确定所述待加锁数据集合对应的数据分段锁;

根据所述待加锁数据集合对应的数据分段锁的唯一标识,将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中。

9. 根据权利要求7所述的装置,其特征在于,所述申请锁模块被配置为:

若所述数据分段锁锁定的数据集合与所述待加锁数据集合存在交集,则进入等待状态;

等待锁定所述交集的数据分段锁被释放之后,申请数据分段锁。

10. 根据权利要求9所述的装置,其特征在于,所述释放锁模块被配置为:

若在预设等待时间内,所述锁定所述交集的数据分段锁未被释放,则释放所述操作锁。

11. 根据权利要求7所述的装置,其特征在于,所述申请锁模块被配置为:

若所述服务器未将所述操作锁分配给所述集群服务节点,则进入等待状态;

等待持有所述操作锁的服务节点释放所述操作锁之后,申请所述操作锁。

12. 一种分布式锁的实现装置,应用于服务器,其特征在于,包括:

接收模块,被配置为接收集群服务节点发送的申请操作锁的请求;

分配模块,被配置为若操作锁未被其他服务节点持有,则将所述操作锁分配给所述集群服务节点;若操作锁被其他服务节点持有,则令所述集群服务节点进入等待状态,等待持有所述操作锁的服务节点释放所述操作锁之后,将所述操作锁分配给所述集群服务节点;

生成模块,被配置为在将所述操作锁分配给所述集群服务节点之后,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,生成所述待加锁数据集合对应的数据分段锁,以通过所述集群服务节点将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中,然后所述集群服务节点释放所述操作锁,获得所述待加锁数据集合对应的数据分段锁,接着所述集群服务节点执行业务操作,在业务操作完成后,将所述待加锁数据集合从所述数据分段锁锁定的数据集合中移除,释放所述待加锁数据集合对应的数据分段锁,其中所述操作锁负责锁定对数据分段的操作。

13. 一种电子设备,其特征在于,包括:

处理器；

用于存储所述处理器可执行指令的存储器；

其中，所述处理器被配置为执行所述指令，以实现如权利要求1至5中任一项所述的分布式锁的实现方法或者如权利要求6所述的分布式锁的实现方法。

14. 一种计算机可读存储介质，当所述计算机可读存储介质中的指令由电子设备的处理器执行时，使得电子设备能够执行如权利要求1至5中任一项所述的分布式锁的实现方法或者如权利要求6所述的分布式锁的实现方法。

分布式锁的实现方法、装置、电子设备及介质

技术领域

[0001] 本公开涉及计算机技术领域,尤其涉及一种分布式锁的实现方法、分布式锁的实现装置、电子设备以及计算机可读存储介质。

背景技术

[0002] 分布式锁是控制分布式系统或不同系统之间共同访问共享资源的一种锁实现,如果不同的系统或同一个系统的不同主机之间共享了某个资源时,往往需要互斥来防止彼此干扰来保证一致性。相关技术中基于Redis实现分布式锁,主要包含加锁和解锁这两个操作,在操作共享资源之前需要先获取锁,操作完毕后需要释放锁。但是,现有的锁只能锁单条记录,无法处理同时锁大批量记录的情况,即无法对大批量数据进行并发操作。

[0003] 需要说明的是,在上述背景技术部分公开的信息仅用于加强对本公开的背景的理解,因此可以包括不构成对本领域普通技术人员已知的现有技术的信息。

发明内容

[0004] 本公开提供一种分布式锁的实现方法、分布式锁的实现装置、电子设备以及计算机可读存储介质,以至少解决相关技术中无法对大批量数据进行并发操作的问题。本公开的技术方案如下:

[0005] 根据本公开实施例的一个方面,提供一种分布式锁的实现方法,应用于集群服务节点,包括:向服务器发送申请操作锁的请求;若所述服务器将操作锁分配给所述集群服务节点,则获取数据分段锁锁定的数据集合;若所述数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中;释放所述操作锁;获得所述待加锁数据集合对应的数据分段锁。

[0006] 在本公开的一个实施例中,将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中,包括:确定所述待加锁数据集合对应的数据分段锁;根据所述待加锁数据集合对应的数据分段锁的唯一标识,将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中。

[0007] 在本公开的一个实施例中,在获得所述待加锁数据集合对应的数据分段锁之后,所述方法还包括:执行业务操作;在业务操作完成后,将所述待加锁数据集合从所述数据分段锁锁定的数据集合中移除,释放所述待加锁数据集合对应的数据分段锁。

[0008] 在本公开的一个实施例中,所述方法还包括:若所述数据分段锁锁定的数据集合与所述待加锁数据集合存在交集,则进入等待状态;等待锁定所述交集的数据分段锁被释放之后,申请数据分段锁。

[0009] 在本公开的一个实施例中,所述方法还包括:若在预设等待时间内所述锁定所述交集的数据分段锁未被释放,则释放所述操作锁。

[0010] 在本公开的一个实施例中,所述方法还包括:若所述服务器未将所述操作锁分配给所述集群服务节点,则进入等待状态;等待持有所述操作锁的服务节点释放所述操作锁

之后,申请所述操作锁。

[0011] 根据本公开实施例的另一方面,提供一种分布式锁的实现方法,应用于服务器,包括:接收集群服务节点发送的申请操作锁的请求;若操作锁未被其他服务节点持有,则将所述操作锁分配给所述集群服务节点;若操作锁被其他服务节点持有,则令所述集群服务节点进入等待状态,等待持有所述操作锁的服务节点释放所述操作锁之后,将所述操作锁分配给所述集群服务节点;以及,在将所述操作锁分配给所述集群服务节点之后,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,生成所述待加锁数据集合对应的数据分段锁。

[0012] 根据本公开实施例的又一方面,提供一种分布式锁的实现装置,应用于集群服务节点,包括:申请锁模块,被配置为向服务器发送申请操作锁的请求;数据获取模块,被配置为若所述服务器将操作锁分配给所述集群服务节点,则获取数据分段锁锁定的数据集合;写入模块,被配置为若所述数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中;释放锁模块,被配置为释放所述操作锁;锁获取模块,被配置为获得所述待加锁数据集合对应的数据分段锁。

[0013] 在本公开的一个实施例中,所述写入模块被配置为:确定所述待加锁数据集合对应的数据分段锁;根据所述待加锁数据集合对应的数据分段锁的唯一标识,将所述待加锁数据集合写入所述数据分段锁锁定的数据集合中。

[0014] 在本公开的一个实施例中,还包括执行模块,被配置为:执行业务操作;以及,所述释放锁模块被配置为:在业务操作完成后,将所述待加锁数据集合从所述数据分段锁锁定的数据集合中移除,释放所述待加锁数据集合对应的数据分段锁。

[0015] 在本公开的一个实施例中,所述申请锁模块被配置为:若所述数据分段锁锁定的数据集合与所述待加锁数据集合存在交集,则进入等待状态;等待锁定所述交集的数据分段锁被释放之后,申请数据分段锁。

[0016] 在本公开的一个实施例中,所述释放锁模块被配置为:若在预设等待时间内,所述锁定所述交集的数据分段锁未被释放,则释放所述操作锁。

[0017] 在本公开的一个实施例中,所述申请锁模块被配置为:若所述服务器未将所述操作锁分配给所述集群服务节点,则进入等待状态;等待持有所述操作锁的服务节点释放所述操作锁之后,申请所述操作锁。

[0018] 根据本公开实施例的又一方面,提供一种分布式锁的实现装置,应用于服务器,包括:接收模块,被配置为接收集群服务节点发送的申请操作锁的请求;分配模块,被配置为若操作锁未被其他服务节点持有,则将所述操作锁分配给所述集群服务节点;若操作锁被其他服务节点持有,则令所述集群服务节点进入等待状态,等待持有所述操作锁的服务节点释放所述操作锁之后,将所述操作锁分配给所述集群服务节点;生成模块,被配置为在将所述操作锁分配给所述集群服务节点之后,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,生成所述待加锁数据集合对应的数据分段锁。

[0019] 根据本公开实施例的再一方面,提供一种电子设备,包括:处理器;用于存储所述处理器可执行指令的存储器;其中,所述处理器被配置为执行所述指令,以实现上述的分布式锁的实现方法。

[0020] 根据本公开实施例的又一方面,提供一种计算机可读存储介质,当所述计算机可

读存储介质中的指令由电子设备的处理器执行时,使得电子设备能够执行上述的分布式锁的实现方法。

[0021] 本公开的实施例提供的技术方案至少带来以下有益效果:在集群服务节点申请操作锁成功的情况下,如果数据分段锁锁定的数据集合和集群服务节点对应的待加锁数据集合的交集为空,通过写入数据的方式锁定待加锁数据集合,即对待加锁数据集合加锁,进而该集群服务节点可以执行业务操作,采用了操作锁和数据分段锁的双重锁操作,可以在多个服务节点对大批量数据进行操作时对数据集合的不同分段加锁,从而可以实现对大批量数据进行并发操作,解决相关技术中无法对大批量数据进行并发操作的问题。

[0022] 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,并不能限制本公开。

附图说明

[0023] 此处的附图被并入说明书中并构成本说明书的一部分,示出了符合本公开的实施例,并与说明书一起用于解释本公开的原理,并不构成对本公开的不当限定。

[0024] 图1是分布式系统的结构示意图;

[0025] 图2是根据一示例性实施例示出的分布式锁的实现方法的流程图;

[0026] 图3是根据一示例性实施例示出的数据分段锁的加锁和释放锁的流程图;

[0027] 图4是根据另一示例性实施例示出的分布式锁的实现方法的流程图;

[0028] 图5是根据又一示例性实施例示出的分布式锁的实现方法的流程图;

[0029] 图6是服务节点与服务器之间的数据交互图;

[0030] 图7是根据一示例性实施例示出的分布式锁的实现装置框图;

[0031] 图8是根据另一示例性实施例示出的分布式锁的实现装置框图;

[0032] 图9是根据一示例性实施例示出的分布式锁的实现设备的结构框图。

具体实施方式

[0033] 为了使本领域普通人员更好地理解本公开的技术方案,下面将结合附图,对本公开实施例中的技术方案进行清楚、完整地描述。

[0034] 需要说明的是,本公开的说明书和权利要求书及上述附图中的术语“第一”、“第二”等是用于区别类似的对象,而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换,以便这里描述的本公开的实施例能够以除了在这里图示或描述的那些以外的顺序实施。以下示例性实施例中所描述的实施方式并不代表与本公开相一致的所有实施方式。相反,它们仅是与如所附权利要求书中所详述的、本公开的一些方面相一致的装置和方法的例子。

[0035] 本公开实施例提供的方法可以由任意类型的电子设备执行,例如服务器或者终端设备,或者服务器和终端设备的交互执行。终端设备以及服务器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。

[0036] 其中,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、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。

[0037] 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。

[0038] 图1是分布式系统的结构示意图。如图1所示,分布式系统包括服务器和集群服务节点。其中,集群服务节点为用于处理数据请求的服务器节点,其数量可以为一个或多个,图1中以两个服务节点为例进行示例说明。每个服务节点可以包括多个服务,每个服务节点均向服务器申请分布式锁,服务器可以看作为用于管理分布式锁的锁管理器。接下来,分别以执行主体为集群服务节点和服务器说明本公开实施例提供的分布式锁的实现方法。

[0039] 图2是根据一示例性实施例示出的分布式锁的实现方法的流程图。如图2所示,该方法应用于集群服务节点,包括以下步骤:

[0040] 步骤S210,向服务器发送申请操作锁的请求;

[0041] 步骤S220,若服务器将操作锁分配给集群服务节点,则获取数据分段锁锁定的数据集合;

[0042] 步骤S230,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则将待加锁数据集合写入数据分段锁锁定的数据集合中;

[0043] 步骤S240,释放操作锁;

[0044] 步骤S250,获得待加锁数据集合对应的数据分段锁。

[0045] 为了便于理解,先对本公开实施例的操作锁和数据分段锁进行说明。操作锁和数据分段锁均为基于Redis实现加锁和解锁的,可以用于图1所示的分布式系统中。集群服务节点进行业务操作之前,需要向服务器申请操作锁和数据分段锁,若申请成功,可以进行业务操作。操作锁负责锁定对Redis数据分段的操作;数据分段锁可以锁定需要操作的数据范围,该数据范围可以用一个key集合表示。具体的,操作锁可以为基于Redis的set命令实现的,key和value为字符串,key可以标记为actionkey.actionkey为操作锁的名称,能够唯一标记操作锁,value为加锁线程的标记。数据分段锁可以为基于Redis的集合命令实现的,key为字符串,此key可以标记为datakey,值是一个字符串集合,记为dataSet.datakey为数据分段锁的名称,能够唯一标记一把数据分段锁,dataSet是处于锁定状态的数据集合。另外,上文提到服务器可以看作为用于管理分布式锁的锁管理器,而操作锁和数据分段锁均为基于Redis实现的,可以认为服务器为Redis服务器。

[0046] 在步骤S210中,集群服务节点向服务器发送申请操作锁的请求,然后服务器读取操作锁的当前状态,判断是否将操作锁分配给该集群服务节点。如果操作锁未被其他服务节点持有,则服务器将操作锁分配给该集群服务节点。如果操作锁被其他服务节点持有,那么服务器不能将操作锁分配给该集群服务节点。此种情况下,该集群服务节点进入等待状态,等待持有操作锁的服务节点释放操作锁之后,可以申请操作锁。举例来说,有第一服务节点和第二服务节点,在第一服务节点向服务器申请操作锁时,服务器将操作锁分配给第二服务节点了,此时第一服务节点需要等待第二服务节点释放操作锁后,再次向服务器申请操作锁。其中,第一服务节点和第二服务节点可以为不同的服务节点;第一服务节点和第二服务节点也可以为相同的服务节点。比如,第一服务节点包括服务1和服务2,在服务1申请到操作锁且未释放的情况下,服务2向服务器申请操作锁,此时需要等服务1释放操作锁之后,服务2可以成功申请操作锁。

[0047] 上文提到数据分段锁可以锁定需要操作的数据范围,或者说,数据分段锁锁定的数据集合为当前操作的数据范围。举例来说,数据分段锁锁定的数据集合中包括数据A1到An,那么可以认为当前在对数据A1到An进行业务处理,如将数据A1到An导入数据库中。在步骤S220中,该集群服务节点申请操作锁成功后,可以获取数据分段锁锁定的数据集合,即获取到当前操作的数据范围。当然,可以是其他服务节点对该数据范围进行业务处理,也可以是该集群服务节点对该数据范围进行业务处理。比如,第一服务节点包括服务1和服务2,服务1在对某数据范围进行业务处理,即数据分段锁锁定的数据集合包括该数据范围,而服务2有业务请求要处理,此时第一服务节点也可以申请操作锁。

[0048] 在步骤S230中,该集群服务节点可以判断当前所有数据分段锁锁定的数据集合(即,当前操作的数据范围)与待加锁数据集合是否存在交集。其中,待加锁数据集合可以认为是此次申请分布式锁需要操作的数据范围,比如第一服务节点要对数据A2到A5进行业务处理,需要申请分布式锁,那么待加锁数据集合为数据A2到A5。

[0049] 如果该集群服务节点判断当前操作的数据范围与待加锁数据集合不存在交集,也就是说,当前操作的数据范围中不包含待加锁数据集合的数据,即当前没有服务节点对此次申请分布式锁需要操作的数据进行业务处理。此种情况下,可以将待加锁数据集合写入数据分段锁锁定的数据集合中,即利用数据分段锁锁定了待加锁数据集合,从而可以实现待加锁数据集合被锁定期间,其他服务节点(即,该集群服务节点之外的节点)不会对待加锁数据集合进行业务处理。

[0050] 还有,在将待加锁数据集合写入数据分段锁锁定的数据集合后,该集群服务节点可以释放操作锁,这样等待操作锁的其他服务节点可以申请到操作锁。接着,该集群服务节点可以获得待加锁数据集合对应的数据分段锁,即该集群服务节点加数据分段锁成功。

[0051] 需要注意的是,本公开实施例中,数据分段锁的数量为一个或多个,数据分段锁锁定的数据集合为当前所有数据分段锁锁定的数据集合。比如,第一服务节点要对数据A2到A5进行业务处理,第二服务节点要对数据A10到A20进行业务处理,可见第一服务节点需要处理的数据范围是与第二服务节点需要处理的数据范围是不同的。如果第一服务节点和第二服务节点均申请锁,那么第一服务节点获得的数据分段锁用于锁定数据A2到A5,第二服务节点获得的数据分段锁用于锁定数据A10到A20,此时数据分段锁的数量为两个。当然,也存在数据分段锁的数量为零的情况,如分布式锁的实现方法的执行初期,服务器还未生成数据分段锁,此种情况下,数据分段锁锁定的数据集合为空。

[0052] 相关技术中无法对大批量数据进行并发操作的问题,本公开的实施例提供的分布式锁的实现方法中,在集群服务节点申请操作锁成功的情况下,如果数据分段锁锁定的数据集合和集群服务节点对应的待加锁数据集合的交集为空,通过写入数据的方式锁定待加锁数据集合,即对待加锁数据集合加锁,进而该集群服务节点可以执行业务操作,采用了操作锁和数据分段锁的双重锁操作,可以在多个服务节点对大批量数据进行操作时对数据集合的不同分段加锁,从而可以实现对大批量数据进行并发操作,解决相关技术中无法对大批量数据进行并发操作的问题。

[0053] 在示例性实施例中,将待加锁数据集合写入数据分段锁锁定的数据集合中,可以包括:确定待加锁数据集合对应的数据分段锁;根据待加锁数据集合对应的数据分段锁的唯一标识,将待加锁数据集合写入数据分段锁锁定的数据集合中。

[0054] 如果数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则服务器会生成待加锁数据集合对应的数据分段锁,该数据分段锁用于锁定待加锁数据集合,该集群服务节点可以确定服务器生成的待加锁数据集合对应的数据分段锁,然后该集群服务节点可以根据确定的数据分段锁的唯一标识,将待加锁数据集合写入数据分段锁锁定的数据集合中。

[0055] 以及,在示例性实施例中,在获得待加锁数据集合对应的数据分段锁之后,该集群服务节点可以执行业务操作,在业务操作完成后,将待加锁数据集合从数据分段锁锁定的数据集合中移除,释放待加锁数据集合对应的数据分段锁。

[0056] 该集群服务节点加数据分段锁成功后,可以进行业务操作,比如将待加锁数据集合对应的数据导入数据库中。在业务操作完成后,该集群服务节点可以将待加锁数据集合从数据分段锁锁定的数据集合中移除,释放待加锁数据集合对应的数据分段锁,这样其他服务节点可以对待加锁数据集合中的数据进行锁定并进行业务操作。

[0057] 图3示出了根据一示例性实施例示出的数据分段锁的加锁和释放锁的流程图。如图3所示,该方法应用于集群服务节点,包括以下步骤:

[0058] 步骤S310,在数据分段锁锁定的数据集合与待加锁数据集合不存在交集的情况下,获取待加锁数据集合对应的数据分段锁;

[0059] 步骤S320,根据待加锁数据集合对应的数据分段锁的唯一标识,将待加锁数据集合写入数据分段锁锁定的数据集合中;

[0060] 步骤S330,释放操作锁;

[0061] 步骤S340,获得待加锁数据集合对应的数据分段锁;

[0062] 步骤S350,执行业务操作;

[0063] 步骤S360,在业务操作完成后,将待加锁数据集合从数据分段锁锁定的数据集合中移除,释放待加锁数据集合对应的数据分段锁。

[0064] 集群服务节点成功申请到操作锁的情况下,可以判断数据分段锁锁定的数据集合与待加锁数据集合是否存在交集。如果不存在交集,则该集群服务节点可以获取到服务器生成的该待加锁数据集合对应的数据分段锁。然后,根据该数据分段锁的唯一标识,将待加锁数据集合写入数据分段锁锁定的数据集合中,即锁定该待加锁数据集合。接着,该集群服务节点可以释放操作锁,释放操作锁之后说明该集群服务节点加数据分段锁成功。这样,该集群服务节点可以进行业务操作,以及在业务操作完成后,将该待加锁数据集合从数据分段锁锁定的数据集合中移除,从而可以释放待加锁数据集合对应的数据分段锁。

[0065] 在数据分段锁的加锁和释放锁的过程中,可以通过将待加锁数据集合写入数据分段锁锁定的数据集合的方式,使得集群服务节点加数据分段锁成功,保证了同一时间只有该集群服务节点对待加锁数据集合中的数据进行业务操作,保证了操作的原子性;并且,在业务操作完成后,将待加锁数据集合从数据分段锁锁定的数据集合中移除,使得该集群服务节点成功释放数据分段锁,以便其他服务节点可以申请到该待加锁数据集合对应的数据分段锁。

[0066] 在示例性实施例中,分布式锁的实现方法还可以包括:若数据分段锁锁定的数据集合与待加锁数据集合存在交集,则进入等待状态;等待锁定交集的数据分段锁被释放之后,申请数据分段锁。

[0067] 如果数据分段锁锁定的数据集合与待加锁数据集合存在交集,则当前操作的数据范围中包含待加锁数据集合的数据,即此次申请分布式锁需要操作的数据正在被某服务节点处理。此种情况下,确定该集群服务节点加数据分段锁失败,需要进入等待状态,等待锁定交集的数据分段锁被释放之后,该集群服务节点申请数据分段锁。比如,待加锁数据集合为数据A1到A5,数据分段锁锁定的数据集合中包含数据A1和A2,那么说明某服务节点在对数据A1和A2进行业务处理。需要等待该服务节点对数据A1和A2进行业务操作之后,将数据A1和A2从锁定的数据集合中移除,即锁定数据A1和A2的数据分段锁被释放,该集群服务节点可以申请数据分段锁。

[0068] 此外,本公开实施例中,若在预设等待时间内,锁定交集的数据分段锁未被释放,则释放操作锁。就是说,如果数据分段锁锁定的数据集合与待加锁数据集合存在交集,则该集群服务节点需要进入等待状态,等待锁定交集的数据分段锁被释放。但是,如果在预设等待时间内,该锁定交集的数据分段锁未被释放,那么有可能是锁定交集的数据分段锁存在问题,也有可能是交集的数据存在问题,也有可能是正在执行的业务操作较复杂。此种情况下,该集群服务节点可以释放操作锁,以便于等待操作锁的其他服务节点可以申请到操作锁。还有,该集群服务节点释放操作锁后,可以加入等待操作锁的队列中,再次申请操作锁。

[0069] 综上,在数据分段锁锁定的数据集合与待加锁数据集合存在交集的情况下,该集群服务节点进入等待状态,等待锁定交集的数据分段锁被释放后,可以申请数据分段锁,进而进行业务操作;以及如果在预设等待时间内锁定交集的数据分段锁未被释放,可以释放操作锁,以便于等待操作锁的进程可以申请到操作锁;还有,在该集群服务节点释放操作锁后,加入等待操作锁的队列中,从而可以再次申请操作锁。

[0070] 图4是根据另一示例性实施例示出的分布式锁的实现方法的流程图。如图4所示,该方法应用于集群服务节点,包括以下步骤:

[0071] 步骤S401,申请操作锁,具体为使用Redis的set命令申请名称为actionkey的操作锁,如果申请成功,则获取到操作锁,如果申请失败,则等待并重试直到超过指定超时时间;

[0072] 步骤S402,在申请到操作锁之后,根据数据分段锁的唯一标识datakey获取锁定的数据集合dataSet,需要获取到所有数据分段锁锁定的数据集合;

[0073] 步骤S403,判断待加锁数据集合waitLockSet和获取的dataSet是否存在交集;

[0074] 步骤S404,如果判断结果为不存在交集,那么获取待加锁数据集合对应的数据分段锁,根据该数据分段锁的唯一标识datakey,使用Redis的SADD命令向Redis的dataSet写入waitLockSet集合的所有值;

[0075] 步骤S405,写入完成之后,释放操作锁,即删除actionkey,释放操作锁之后,确定加数据分段锁成功;

[0076] 步骤S406,加数据分段锁成功后进行业务操作,比如将待加锁数据集合对应的数据导入数据库中;

[0077] 步骤S407,在业务操作完成后,释放数据分段锁,使用Redis的SREM命令将waitLockSet从datakey的值集合dataSet中移除;

[0078] 步骤S408,如果判断结果为存在交集,确定本次加数据分段锁失败,需要进入等待状态;

[0079] 步骤S409,如果在预设等待时间内,锁定交集的数据分段锁被释放,申请数据分段

锁；

[0080] 步骤S410,如果在预设等待时间内,锁定交集的数据分段锁未被释放,释放操作锁,以及加入等待操作锁的队列中,从而可以再次申请操作锁。

[0081] 在集群服务节点申请操作锁成功的情况下,如果数据分段锁锁定的数据集合和集群服务节点对应的待加锁数据集合的交集为空,通过写入数据的方式锁定待加锁数据集合,即对待加锁数据集合加锁,进而该集群服务节点可以执行业务操作,采用了操作锁和数据分段锁的双重锁操作,可以在多个服务节点对大批量数据进行操作时对数据集合的不同分段加锁,从而可以实现对大批量数据进行并发操作,解决相关技术中无法对大批量数据进行并发操作的问题。以及,在判断数据分段锁锁定的数据集合和待加锁数据集合存在交集的情况下,集群服务节点进入等待状态,等待锁定交集的数据分段锁被释放后,可以申请数据分段锁,进而进行业务操作;并且,如果在预设等待时间内锁定交集的数据分段锁未被释放,可以释放操作锁,以便于等待操作锁的其他服务节点可以申请到操作锁,还有在集群服务节点释放操作锁后,加入等待操作锁的队列中,从而可以再次申请操作锁。

[0082] 接下来以执行主体为服务器说明本公开实施例的分布式锁的实现方法。图5是根据又一示例性实施例示出的分布式锁的实现方法的流程图。如图5所示,该方法应用于服务器,包括以下步骤:

[0083] 步骤S510,接收集群服务节点发送的申请操作锁的请求;

[0084] 步骤S520,若操作锁未被其他服务节点持有,则将操作锁分配给该集群服务节点;

[0085] 步骤S530,若操作锁被其他服务节点持有,则令该集群服务节点进入等待状态,等待持有所述操作锁的服务节点释放操作锁之后,将操作锁分配给该集群服务节点;

[0086] 步骤S540,在将操作锁分配给该集群服务节点之后,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,生成待加锁数据集合对应的数据分段锁。

[0087] 在上文中已经说明,服务器可以看作为用于管理分布式锁的锁管理器,即服务器用于管理操作锁和数据分段锁。服务器可以接收到各集群服务节点发送的申请操作锁的请求,为了便于理解,以第一服务节点为例进行说明。如果操作锁没有被其他服务节点(第一服务节点外的节点)持有,那么服务器可以将操作锁分配给第一服务节点,即第一服务节点加操作锁成功。如果操作锁被其他服务节点占用,那么服务器可以令第一服务节点进入等待状态,在其他服务节点释放操作锁之后,可以将操作锁分配给第一服务节点。还有,如果存在多个服务节点在等待操作锁,那么可以按照等待顺序分配操作锁。

[0088] 在服务器将操作锁分配给集群服务节点之后,该集群服务节点可以判断数据分段锁锁定的数据集合与待加锁数据集合是否存在交集,具体的判断过程在上文中已经说明,此处不再累述。如果确定数据分段锁锁定的数据集合与待加锁数据集合不存在交集,那么服务器可以生成待加锁数据集合对应的数据分段锁。然后,该集群服务节点可以:确定该待加锁数据集合对应的数据分段锁;根据待加锁数据集合对应的数据分段锁的唯一标识,将待加锁数据集合写入数据分段锁锁定的数据集合中;释放操作锁,获得待加锁数据集合对应的数据分段锁,即该集群服务节点加数据分段锁成功;执行业务操作,在业务操作完成后,将待加锁数据集合从数据分段锁锁定的数据集合中移除,释放待加锁数据集合对应的数据分段锁。

[0089] 本公开实施例提供的分布式锁的实现方法中,服务器用于管理操作锁和数据分段

锁,具体的,在操作锁未被占用时将操作锁分配给服务节点,以及在服务节点确定数据分段锁锁定的数据集合与待加锁数据集合不存在交集的情况下,生成待加锁数据集合对应的数据分段锁,进而可以利用生成的数据分段锁对服务节点加锁,以便服务节点进行业务操作,采用了操作锁和数据分段锁的双重锁操作,可以在多个服务节点对大批量数据进行操作时对数据集合的不同分段加锁,从而可以实现对大批量数据进行并发操作,解决相关技术中无法对大批量数据进行并发操作的问题。

[0090] 图6是服务节点与服务器之间的数据交互图。参考图6,第一服务节点和第二服务节点向服务器发送申请操作锁的请求。第一服务节点申请操作锁成功,即服务器将操作锁分配给第一服务节点;第一服务节点获取数据分段锁锁定的数据集合,判断锁定的数据集合与待加锁数据集合是否存在交集;如果不存在交集,服务器生成待加锁数据集合对应的数据分段锁,第一服务节点获取到生成的数据分段锁;第一服务节点根据生成的数据分段锁的唯一标识,将待加锁数据集合写入数据分段锁锁定的数据集合中,即锁定待加锁数据集合;第一服务节点释放操作锁,确定加数据分段锁成功;第一服务节点进行业务操作,在业务操作完成后,将待加锁数据集合从数据分段锁锁定的数据集合中移除,释放待加锁数据集合对应的数据分段锁。从图6还可以看出,第二服务节点申请操作锁失败,那么第二服务节点进入等待状态,等待第一服务节点释放操作锁之后,再次申请操作锁。

[0091] 图6所示的为第一服务节点判断数据分段锁锁定的数据集合与待加锁数据集合不存在交集的情况,图6未示出数据分段锁锁定的数据集合与待加锁数据集合存在交集的情况。如果第一服务节点判断数据分段锁锁定的数据集合与待加锁数据集合存在交集,第一服务节点需要进入等待状态,等待锁定交集的数据分段锁被释放。如果在预设等待时间内锁定交集的数据分段锁被释放,第一服务节点可以申请数据分段锁,进而进行业务操作。如果在预设等待时间内锁定交集的数据分段锁未被释放,第一服务节点可以释放操作锁,以便于等待操作锁的第二服务节点可以申请到操作锁。此外,第一服务节点释放操作锁后,可以加入等待操作锁的队列中,再次申请操作锁。

[0092] 图7是根据一示例性实施例示出的分布式锁的实现装置框图。图7所示的分布式锁的实现装置700应用于集群服务节点。参照图7,该装置包括:申请锁模块710、数据获取模块720、写入模块730、释放锁模块740和锁获取模块750。

[0093] 该申请锁模块710被配置为向服务器发送申请操作锁的请求。

[0094] 该数据获取模块720被配置为若服务器将操作锁分配给集群服务节点,则获取数据分段锁锁定的数据集合。

[0095] 该写入模块730被配置为若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则将待加锁数据集合写入数据分段锁锁定的数据集合中。

[0096] 该释放锁模块740被配置为释放操作锁。

[0097] 该锁获取模块750被配置为获得待加锁数据集合对应的数据分段锁。

[0098] 在示例性实施例中,该写入模块730还被配置为:确定待加锁数据集合对应的数据分段锁;根据待加锁数据集合对应的数据分段锁的唯一标识,将待加锁数据集合写入数据分段锁锁定的数据集合中。

[0099] 在示例性实施例中,图7所示的分布式锁的实现装置700还可以包括执行模块760。该执行模块760被配置为:执行业务操作。以及,该释放锁模块740被配置为:在业务操作完

成后,将待加锁数据集合从数据分段锁锁定的数据集合中移除,释放待加锁数据集合对应的数据分段锁。

[0100] 在示例性实施例中,该申请锁模块710还被配置为:若数据分段锁锁定的数据集合与待加锁数据集合存在交集,则进入等待状态;等待锁定交集的数据分段锁被释放之后,申请数据分段锁。

[0101] 在示例性实施例中,该释放锁模块740还被配置为:若在预设等待时间内,锁定交集的数据分段锁未被释放,则释放操作锁。

[0102] 在示例性实施例中,该申请锁模块710还被配置为:若服务器未将操作锁分配给集群服务节点,则进入等待状态;等待持有操作锁的服务节点释放操作锁之后,申请操作锁。

[0103] 关于上述实施例中的装置,其中各个模块执行操作的具体方式已经在有关该方法的实施例中进行了详细描述,此处将不做详细阐述说明。

[0104] 图8是根据另一示例性实施例示出的分布式锁的实现装置框图。图8所示的分布式锁的实现装置800应用于服务器。参照图8,该装置包括:接收模块810、分配模块820和生成模块830。

[0105] 该接收模块810被配置为接收集群服务节点发送的申请操作锁的请求。

[0106] 该分配模块820被配置为若操作锁未被其他服务节点持有,则将操作锁分配给集群服务节点;若操作锁被其他服务节点持有,则令集群服务节点进入等待状态,等待持有操作锁的服务节点释放操作锁之后,将操作锁分配给集群服务节点。

[0107] 该生成模块830被配置为在将操作锁分配给集群服务节点之后,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,生成待加锁数据集合对应的数据分段锁。

[0108] 关于上述实施例中的装置,其中各个模块执行操作的具体方式已经在有关该方法的实施例中进行了详细描述,此处将不做详细阐述说明。

[0109] 图9是根据一示例性实施例示出的分布式锁的实现设备的结构框图。需要说明的是,图示出的电子设备仅仅是一个示例,不应对本发明实施例的功能和使用范围带来任何限制。

[0110] 下面参照图9来描述根据本发明的这种实施方式的电子设备900。图9显示的电子设备900仅仅是一个示例,不应对本发明实施例的功能和使用范围带来任何限制。

[0111] 如图9所示,电子设备900以通用计算设备的形式表现。电子设备900的组件可以包括但不限于:上述至少一个处理单元910、上述至少一个存储单元920、连接不同系统组件(包括存储单元920和处理单元910)的总线930。

[0112] 其中,所述存储单元存储有程序代码,所述程序代码可以被所述处理单元910执行,使得所述处理单元910执行本说明书上述“示例性方法”部分中描述的根据本发明各种示例性实施方式的步骤。例如,所述处理单元910可以执行如图2中所示的步骤S210,向服务器发送申请操作锁的请求;步骤S220,若服务器将操作锁分配给集群服务节点,则获取数据分段锁锁定的数据集合;步骤S230,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,则将待加锁数据集合写入数据分段锁锁定的数据集合中;步骤S240,释放操作锁;步骤S250,获得待加锁数据集合对应的数据分段锁。再如,所述处理单元910可以执行如图5中所示的步骤S510,接收集群服务节点发送的申请操作锁的请求;步骤S520,若操作锁未被其他服务节点持有,则将操作锁分配给该集群服务节点;步骤S530,若操作锁被其他服务节

点持有,则令该集群服务节点进入等待状态,等待持有所述操作锁的服务节点释放操作锁之后,将操作锁分配给该集群服务节点;步骤S540,在将操作锁分配给该集群服务节点之后,若数据分段锁锁定的数据集合与待加锁数据集合不存在交集,生成待加锁数据集合对应的数据分段锁。

[0113] 存储单元920可以包括易失性存储单元形式的可读介质,例如随机存取存储单元(RAM)9201和/或高速缓存存储单元9202,还可以进一步包括只读存储单元(ROM)9203。

[0114] 存储单元920还可以包括具有一组(至少一个)程序模块9205的程序/实用工具9204,这样的程序模块9205包括但不限于:操作系统、一个或者多个应用程序、其它程序模块以及程序数据,这些示例中的每一个或某种组合中可能包括网络环境的实现。

[0115] 总线930可以为表示几类总线结构中的一种或多种,包括存储单元总线或者存储单元控制器、外围总线、图形加速端口、处理单元或者使用多种总线结构中的任意总线结构的局域总线。

[0116] 电子设备900也可以与一个或多个外部设备1000(例如键盘、指向设备、蓝牙设备等)通信,还可与一个或者多个使得用户能与该电子设备900交互的设备通信,和/或与使得该电子设备900能与一个或多个其它计算设备进行通信的任何设备(例如路由器、调制解调器等等)通信。这种通信可以通过输入/输出(I/O)接口950进行。并且,电子设备900还可以通过网络适配器990与一个或者多个网络(例如局域网(LAN),广域网(WAN)和/或公共网络,例如因特网)通信。如图所示,网络适配器940通过总线930与电子设备900的其它模块通信。应当明白,尽管图中未示出,可以结合电子设备900使用其它硬件和/或软件模块,包括但不限于:微代码、设备驱动器、冗余处理单元、外部磁盘驱动阵列、RAID系统、磁带驱动器以及数据备份存储系统等。

[0117] 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,根据本公开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、终端装置、或者网络设备等)执行根据本公开实施方式的方法。

[0118] 在本公开的示例性实施例中,还提供了一种计算机可读存储介质,其上存储有能够实现本说明书上述方法的程序产品。在一些可能的实施方式中,本发明的各个方面还可以实现为一种程序产品的形式,其包括程序代码,当所述程序产品在终端设备上运行时,所述程序代码用于使所述终端设备执行本说明书上述“示例性方法”部分中描述的根据本发明各种示例性实施方式的步骤。

[0119] 根据本发明实施方式的用于实现上述方法的程序产品,其可以采用便携式紧凑盘只读存储器(CD-ROM)并包括程序代码,并可以在终端设备,例如个人电脑上运行。然而,本发明的程序产品不限于此,在本文件中,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。

[0120] 所述程序产品可以采用一个或多个可读介质的任意组合。可读介质可以是可读信号介质或者可读存储介质。可读存储介质例如可以为但不限于电、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。可读存储介质的更具体的例子(非穷举

的列表)包括:具有一个或多个导线的电连接、便携式盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。

[0121] 计算机可读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,其中承载了可读程序代码。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。可读信号介质还可以是可读存储介质以外的任何可读介质,该可读介质可以发送、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。

[0122] 可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于无线、有线、光缆、RF等等,或者上述的任意合适的组合。

[0123] 可以以一种或多种程序设计语言的任意组合来编写用于执行本发明操作的程序代码,所述程序设计语言包括面向对象的程序设计语言—诸如Java、C++等,还包括常规的过程式程序设计语言—诸如“C”语言或类似的设计语言。程序代码可以完全地在用户计算设备上执行、部分地在用户设备上执行、作为一个独立的软件包执行、部分在用户计算设备上部分在远程计算设备上执行、或者完全在远程计算设备或服务器上执行。在涉及远程计算设备的情形中,远程计算设备可以通过任意种类的网络,包括局域网(LAN)或广域网(WAN),连接到用户计算设备,或者,可以连接到外部计算设备(例如利用因特网服务提供商来通过因特网连接)。

[0124] 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,根据本公开的实施方式,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。

[0125] 此外,尽管在附图中以特定顺序描述了本公开中方法的各个步骤,但是,这并非要求或者暗示必须按照该特定顺序来执行这些步骤,或是必须执行全部所示的步骤才能实现期望的结果。附加的或备选的,可以省略某些步骤,将多个步骤合并为一个步骤执行,以及/或者将一个步骤分解为多个步骤执行等。

[0126] 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,根据本公开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、移动终端、或者网络设备等)执行根据本公开实施方式的方法。

[0127] 本领域技术人员在考虑说明书及实践这里公开的发明后,将容易想到本公开的其他实施方案。本申请旨在涵盖本公开的任何变型、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公开的一般性原理并包括本公开未公开的本技术领域中的公知常识或惯用技术手段。说明书和实施例仅被视为示例性的,本公开的真正范围和精神由所附的权利要求指出。

[0128] 应当理解的是,本公开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。本公开的范围仅由所附的权利要求来限制。

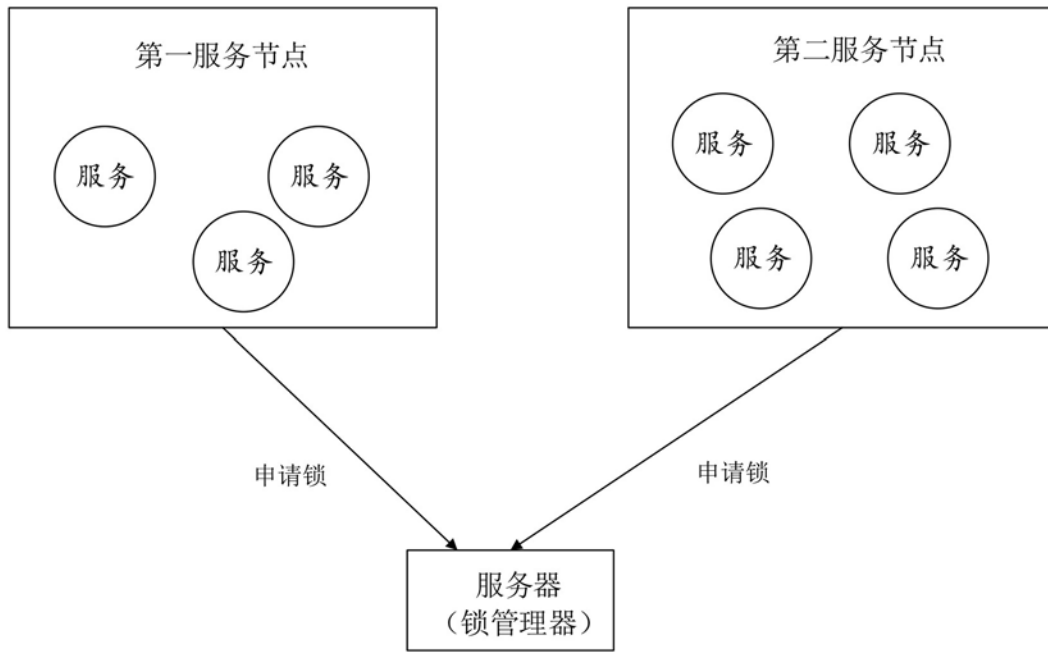


图1

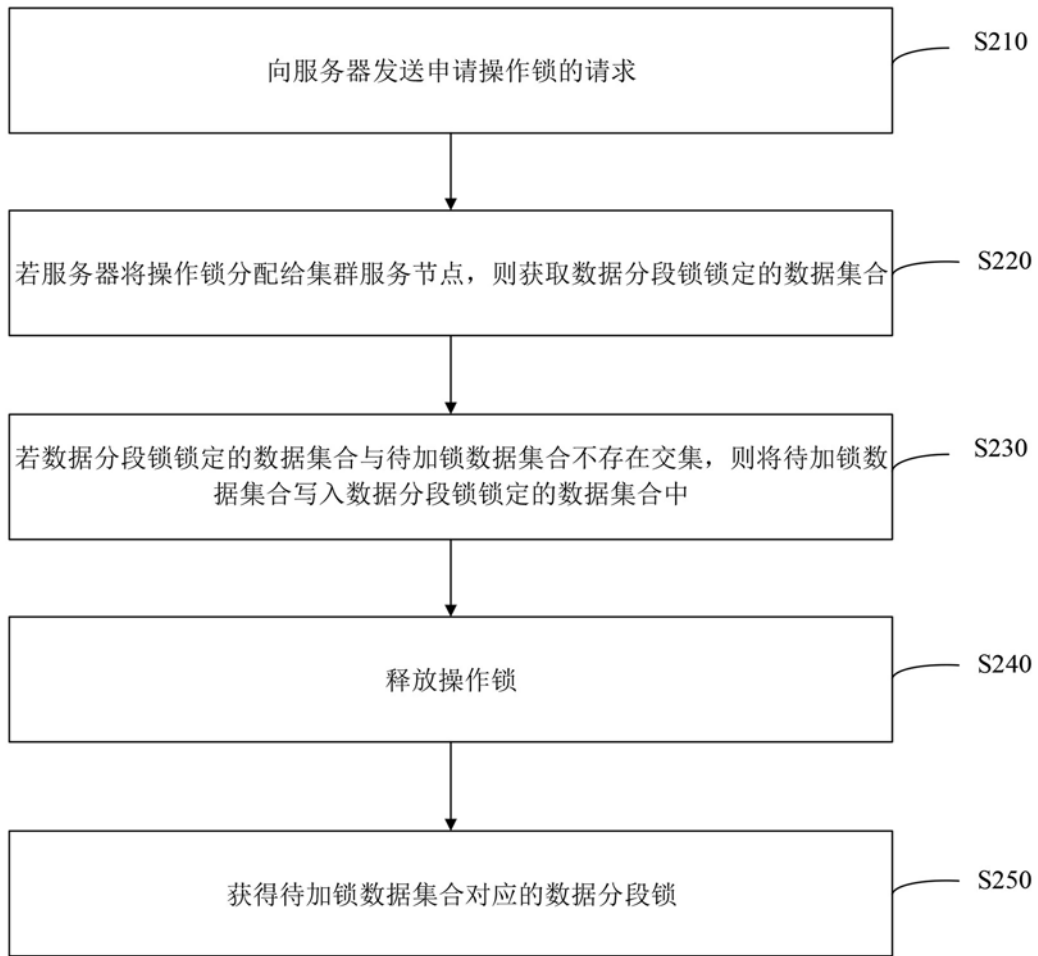


图2

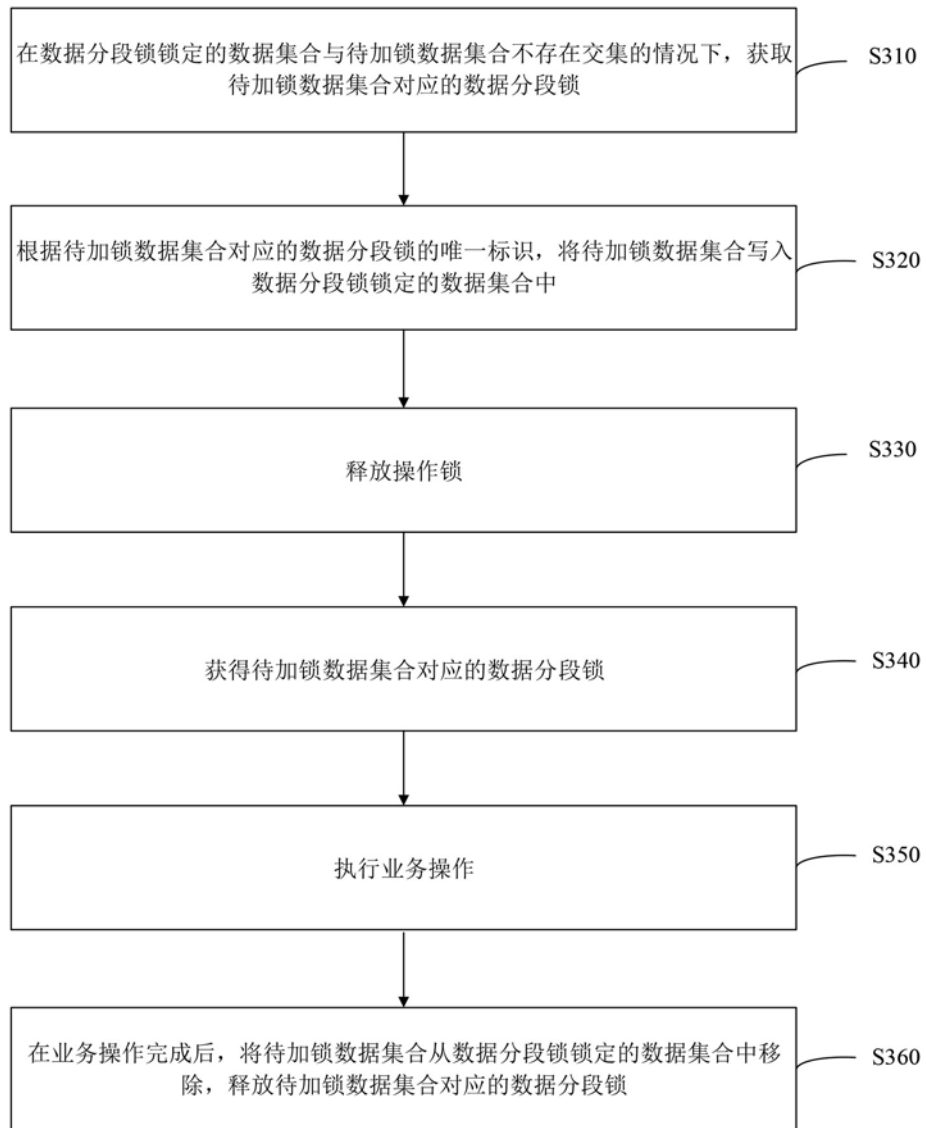


图3

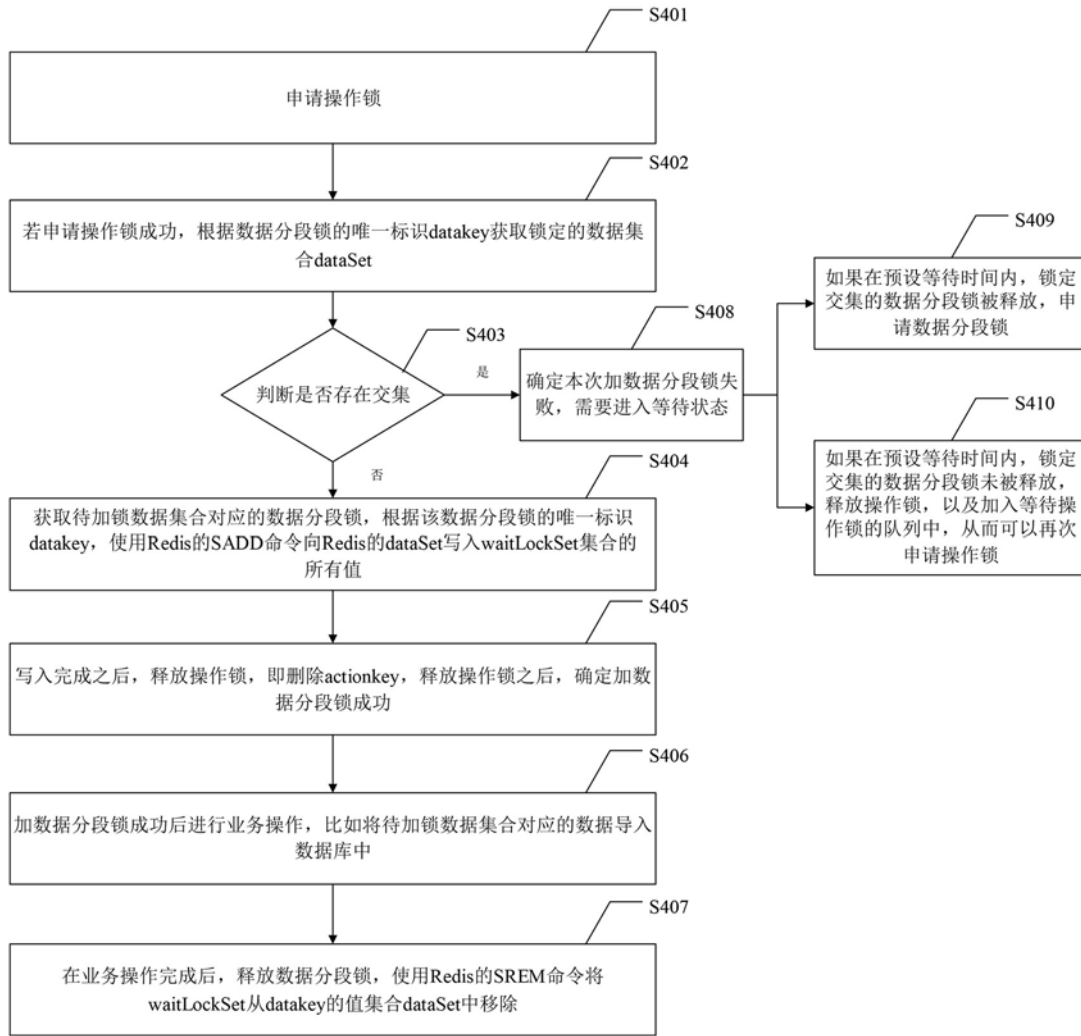


图4

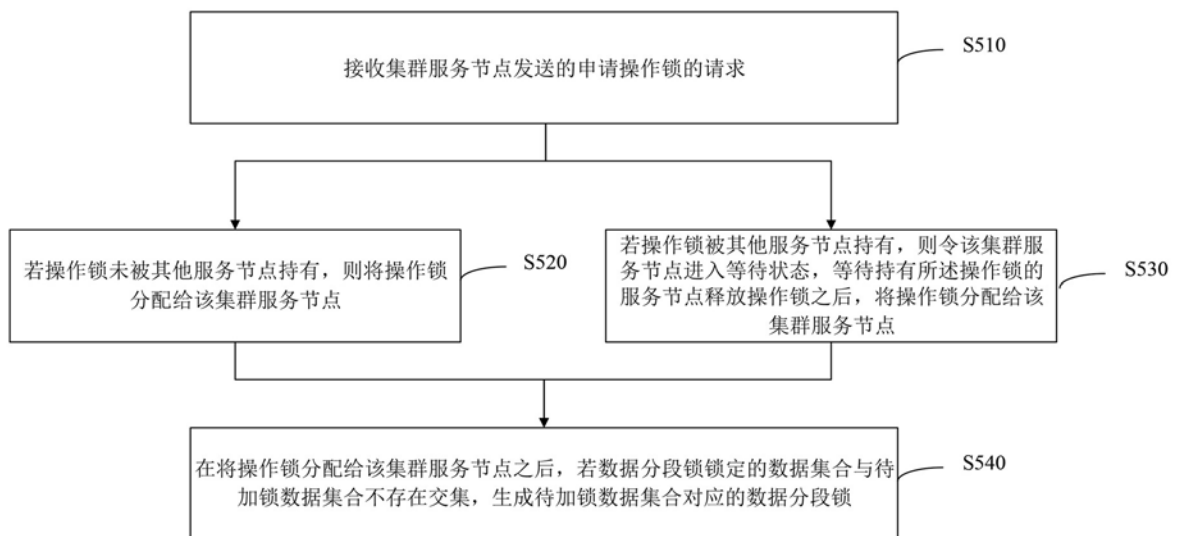


图5

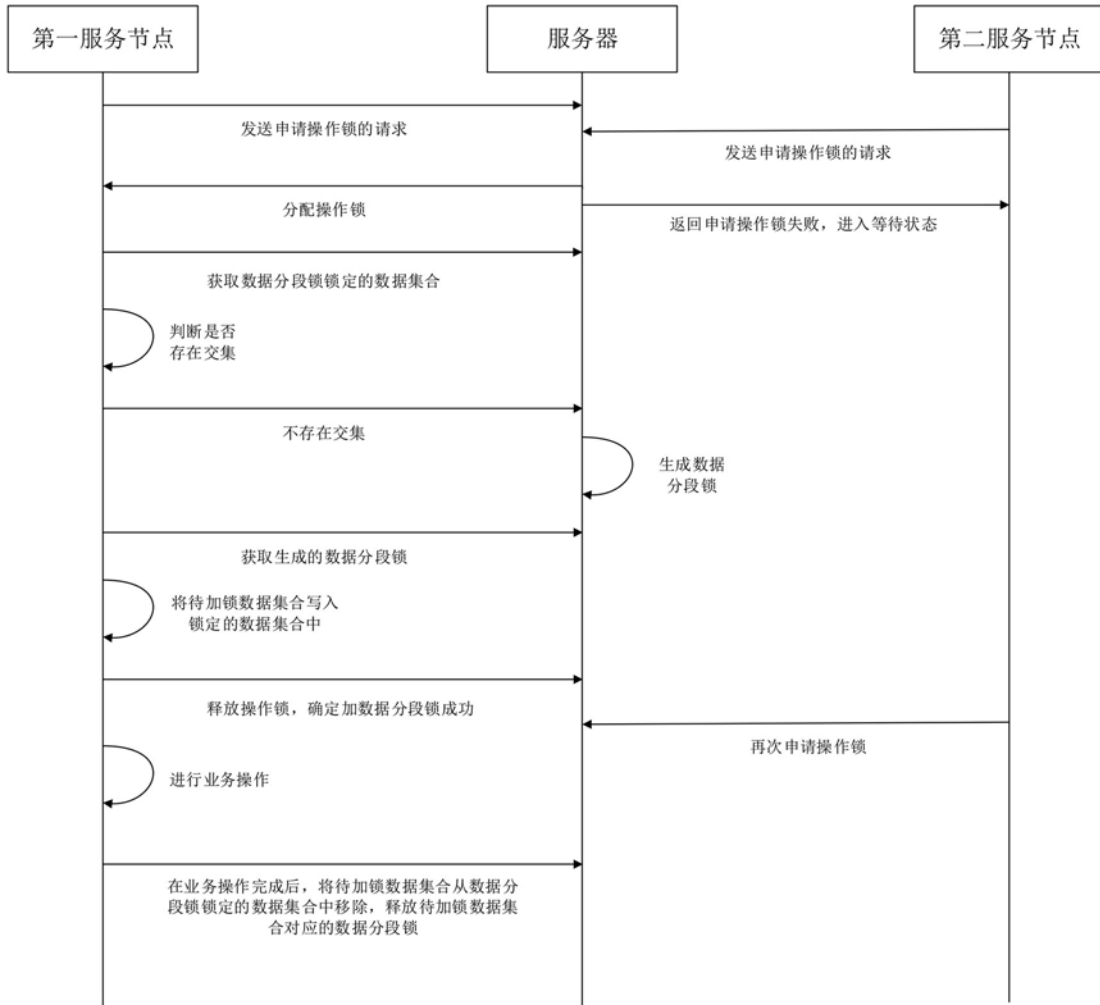


图6

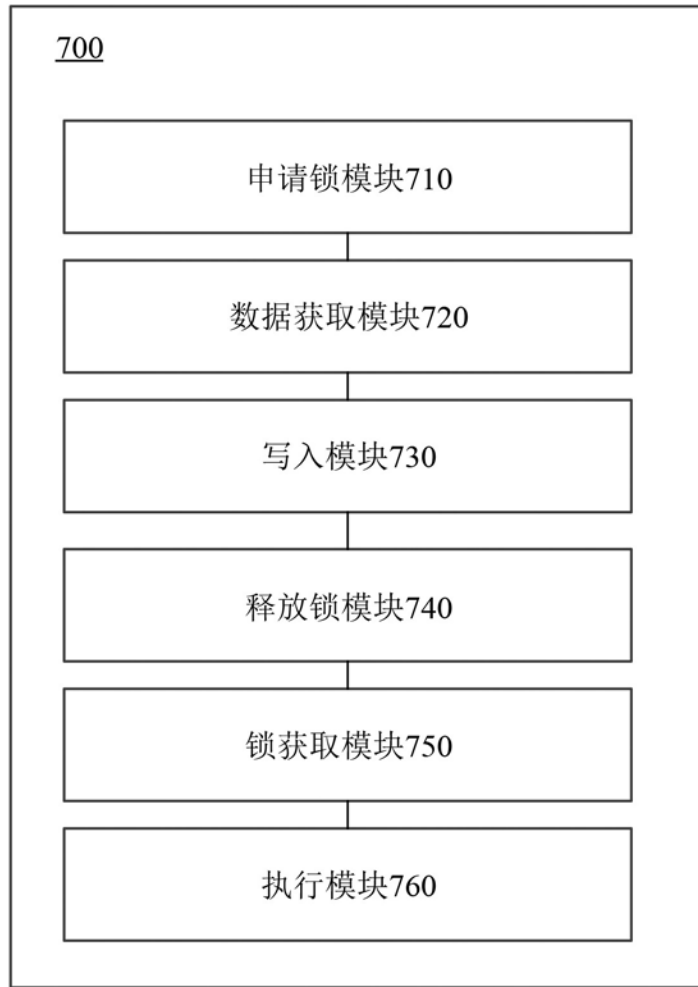


图7

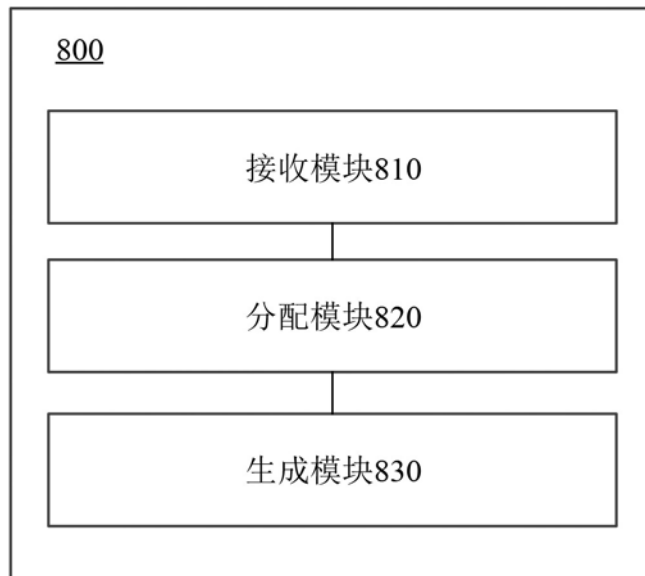


图8

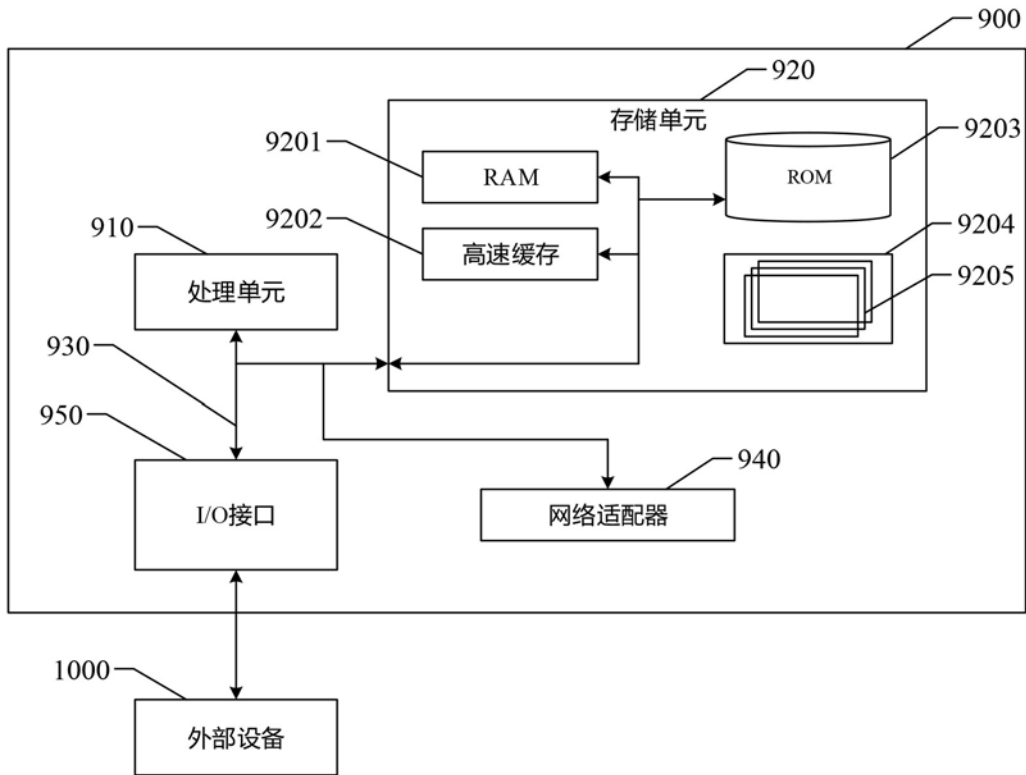


图9