(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 111708494 A
(43)申请公布日 2020.09.25

(21)申请号 202010551837.0

(22)申请日 2020.06.17

(71)申请人 浪潮云信息技术股份公司
地址 250100 山东省济南市高新区浪潮路1036号浪潮科技园S01号楼

(72)发明人 祝乃国

(74)专利代理机构 济南信达专利事务所有限公司 37100
代理人 姜明

(51)Int.Cl.
G06F 3/06(2006.01)
G06F 16/172(2019.01)
G06F 16/182(2019.01)

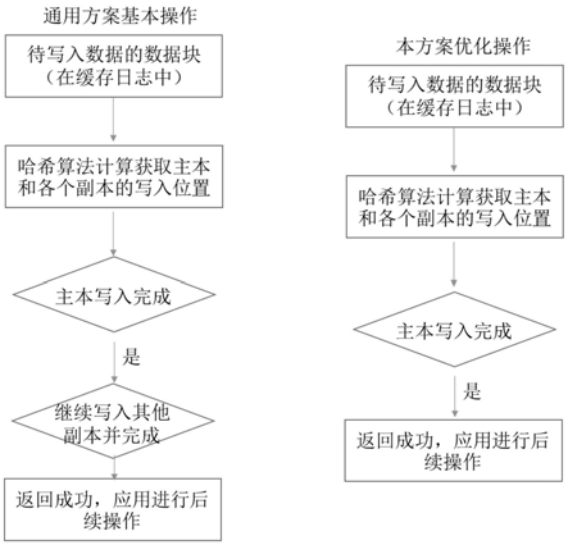
权利要求书2页 说明书5页 附图2页

(54)发明名称

一种分布式存储QOS的实现方法

(57)摘要

本发明涉及分布式存储领域，具体提供了一种分布式存储QOS的实现方法，该方法的步骤为，S01、分布式存储运行性能体系建立及获取；S02、分布式存储QOS实现；S03、分布式存储CAP均衡实现。与现有技术相比，本发明硬件介质确定的前提下，改善了分布式存储的整体性能，并且可通过客户、应用等条件控制读写速率，提高用户的使用体验和感受，提升用户满意度，具有良好的推广价值。



CN 111708494 A

1.一种分布式存储QOS的实现方法,其特征在于,该方法的步骤为,

S01、分布式存储运行性能体系建立及获取;

S02、分布式存储QOS实现;

S03、分布式存储CAP均衡实现。

2.根据权利要求1所述的一种分布式存储QOS的实现方法,其特征在于,在步骤S01中,通过在分布式存储软件中增加原子单元的读写操作记录,结合原子单元所属的文件或归属的其他粒度,统计出冷热文件、原子单元操作频次和发生容量,并将这些指标应用到步骤S02中。

3.根据权利要求2所述的一种分布式存储QOS的实现方法,其特征在于,所述原子单元的操作记录包括标志、时间和耗时长。

4.根据权利要求1所述的一种分布式存储QOS的实现方法,其特征在于,在步骤S02中,包括:

S021、排定各待处理文件的优先级;

S022、按照优先级执行顺序操作。

5.根据权利要求4所述的一种分布式存储QOS的实现方法,其特征在于,在步骤S021前,首先,将文件按照系统缺省设置的读写块大小拆分为等大小的数据块,最后一个可以为不是等大小的数据块;

然后,通过哈希算法获得每个数据块写入磁盘的位置;

最后,在磁盘上执行写入或读取。

6.根据权利要求5所述的一种分布式存储QOS的实现方法,其特征在于,在步骤S021中,对输入的相关信息计算获得一个优先级数字,所述数字与后续调用一起传递给最终执行程序。

7.根据权利要求6所述的一种分布式存储QOS的实现方法,其特征在于,在步骤S022中,排队队列的待处理任务进行优先级从大到小排序,若优先级一样则任务提交时间久的优先。

8.根据权利要求1所述的一种分布式存储QOS的实现方法,其特征在于,在步骤S03中,增加中间状态变量Flag,其过程表现为:ReqNode-1AckNode-1{FlagNode-2,,FlagNode-n},Flag由系统记录并管理,在对应的位置完成后给系统发送AckNode-n通知,则设置FlagNode-n为FinishNode-n。系统另外设置进程检查,当前队列里Flag的情况,对于长时间未完成标志更新的判断其执行情况,如果数据操作进程活动,则提高其执行优先级;如果进程已经退出,则重新发起同步操作。

9.根据权利要求8所述的一种分布式存储QOS的实现方法,其特征在于,在副本异步写入操作过程中,有数据块写入操作发生时,判断队列中是否写入操作完成,若完成,根据缓存中待写入副本的数据块形成标志队列,形成队列中数据块对应标志及操作排序的优先级权重,副本写入操作完成,更新队列操作日志,并删除队列中排队的数据块;

若未完成,则修改优先级的权重,形成队列中数据块对应标志及操作排序的优先级权重,将副本写入操作完成,更新队列操作日志,并删除队列中排队的数据块。

10.根据权利要求9所述的一种分布式存储QOS的实现方法,其特征在于,当副本写入队列维护巡检时,若存在操作异常及等待时间长的数据块,则修改优先级权重,形成队列中数

据块对应标志及操作排序的优先级权重，副本写入操作完成，更新队列操作日志，并删除队列中排队的数据块。

一种分布式存储QOS的实现方法

技术领域

[0001] 本发明涉及分布式存储领域，具体提供一种分布式存储QoS的实现方法。

背景技术

[0002] 分布式存储是相对集中存储提出的概念，就是以大容量硬盘的服务器为存储介质，通过软件管理形成存储供用户使用。分布式存储可以提供各种访问协议，可以支持块和对象等存储模式。

[0003] 分布式存储的基础是通过分区容错来提升数据持久化的稳定性，如在三副本情况下理论计算可以到9个9的存储稳定性，并且为了保持这个稳定性的可用一般使用强一致性来保障。

[0004] 所谓多副本以及强一致性，都以分布式存储的数据读写原理为基础。一般分布式存储要存储一份数据，对用户来说是通过文件形式来表示。而要存储文件，分布式存储一般会把文件按照设置的读写数据块大小进行切分，形成数据单元。利用一定的算法计算这些数据单元可以存储的位置点，然后进行读写即可完成数据的持久化存储。

[0005] 在三副本处理工程中，存在以下缺点：

[0006] 上层应用在主副本全部完成时才会收到结束的标识，传输线路长；

[0007] 都有自动分配及故障时恢复机制，但都基于数据单元本身，没有优先级顺序；

[0008] 如果3T的硬盘损坏，要做到数据重新一致性完成，测试需耗时4小时以上，在这个过程里读写会受影响。

[0009] 副本数一般为奇数，选举时容易出现脑裂，无法确定主本。

[0010] 从以上过程可以看出，分布式存储作为一种存储的技术模式，与传统的集中存储相比，在性能上仍有差距，所以优化并提升分布式存储整体性能是本领域技术人员亟待解决的问题。

发明内容

[0011] 本发明是针对上述现有技术的不足，提供一种实用性强的分布式存储QoS 的实现方法。

[0012] 本发明解决其技术问题所采用的技术方案是：

[0013] 一种分布式存储QOS的实现方法，该方法的步骤为，

[0014] S01、分布式存储运行性能体系建立及获取；

[0015] S02、分布式存储QOS实现；

[0016] S03、分布式存储CAP均衡实现。

[0017] 进一步的，在步骤S01中，通过在分布式存储软件中增加原子单元的读写操作记录，结合原子单元所属的文件或归属的其他粒度，统计出冷热文件、原子单元操作频次和发生容量，并将这些指标应用到步骤S02中。

[0018] 作为优选，所述原子单元的操作记录包括标志、时间和耗时长。

[0019] 进一步的，在步骤S02中，包括：

[0020] S021、排定各待处理文件的优先级；

[0021] S022、按照优先级执行顺序操作。

[0022] 进一步的，在步骤S021前，首先，将文件按照系统缺省设置的读写块大小拆分为等大小的数据块，最后一个可以为不是等大小的数据块；

[0023] 然后，通过哈希算法获得每个数据块写入磁盘的位置；

[0024] 最后，在磁盘上执行写入或读取。

[0025] 进一步的，在步骤S021中，对输入的相关信息计算获得一个优先级数字，所述数字与后续调用一起传递给最终执行程序。

[0026] 进一步的，在步骤S022中，排队队列的待处理任务进行优先级从大到小排序，若优先级一样则任务提交时间久的优先。

[0027] 进一步的，在步骤S03中，增加中间状态变量Flag，其过程表现为：ReqNode-1 AckNode-1{FlagNode-2，，FlagNode-n}，Flag由系统记录并管理，在对应的位置完成后给系统发送AckNode-n通知，则设置FlagNode-n为 FinishNode-n。系统另外设置进程检查，当前队列里Flag的情况，对于长时间未完成标志更新的判断其执行情况，如果数据操作进程活动，则提高其执行优先级；如果进程已经退出，则重新发起同步操作。

[0028] 进一步的，在副本异步写入操作过程中，有数据块写入操作发生时，判断队列中是否写入操作完成，若完成，根据缓存中待写入副本的数据块形成标志队列，形成队列中数据块对应标志及操作排序的优先级权重，副本写入操作完成，更新队列操作日志，并删除队列中排队的数据块；

[0029] 若未完成，则修改优先级的权重，形成队列中数据块对应标志及操作排序的优先级权重，将副本写入操作完成，更新队列操作日志，并删除队列中排队的数据块。

[0030] 进一步的，当副本写入队列维护巡检时，若存在操作异常及等待时间长的数据块，则修改优先级权重，形成队列中数据块对应标志及操作排序的优先级权重，副本写入操作完成，更新队列操作日志，并删除队列中排队的数据块。

[0031] 本发明的一种分布式存储QoS的实现方法和现有技术相比，具有以下突出的有益效果：

[0032] 本发明在分布式存储现有性能水平的前提下，传输线路变短，加入优先级，提高用户的使用体验和感受，提升用户满意度。具体表现在：

[0033] 1)在故障恢复时，通过QoS可以实现有差别的对待，即高优先级的先恢复，改变了分布式存储对数据一视同仁的技术做法。一般3T的磁盘在万兆网络环境里恢复整盘的数据大概需要3-4个小时，再次期间用户的操作请求会受到很大影响。在无法改变整盘恢复时长的前提下，通过QoS模式，可以尽量减少有关用户的等待时间，给予重要用户或应用无影响的使用感受。

[0034] 2)在通用分布式存储的系统中，CAP理论成立，并且考虑了强一致性要素，导致用户使用感受差。本发明在确保数据一致性的基础上，从根本上改变了用户的可用性体验，并且不受副本数据的影响，即副本数据增加不会改变用户可用性体验。

[0035] 通过本发明的方法，在发挥分布式存储优点的同时，尽量改善了其缺点，使性价比有了大幅度提升，促进了在云计算中更广泛的使用。

附图说明

[0036] 为了更清楚地说明本发明实施例或现有技术中的技术方案，下面将对实施例或现有技术描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图是本发明的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

[0037] 附图1是一种分布式存储QoS的实现方法与现有技术的对比图；

[0038] 附图2是本发明副本异步写入操作过程的流程图；

[0039] 附图3是本发明QoS保障时操作处理过程流程图。

具体实施方式

[0040] 为了使本技术领域的人员更好的理解本发明的方案，下面结合具体的实施方式对本发明作进一步的详细说明。显然，所描述的实施例仅仅是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例都属于本发明保护的范围。

[0041] 下面给出一个最佳实施例：

[0042] 对于分布式存储，分区容错是基本要求，否则就失去了价值。因此在设计分布式存储的时候，是在一致性和可用性之间取一个平衡。

[0043] 对于大多数WEB应用，其实并不需要强一致性，因此牺牲一致性而换取高可用性，是多数分布式数据库产品的方向。当然，牺牲一致性，并不是完全不管数据的一致性，否则数据是混乱的，那么系统可用性再高分布式再好也没有了价值。牺牲一致性，只是不再要求关系型数据库中的强一致性，而是只要系统能达到最终一致性即可，考虑到客户体验，这个最终一致的时间窗口，要尽可能的对用户透明，也就是需要保障“用户感知到的一致性”。

[0044] 通过数据的多份异步复制来实现系统的高可用和数据的最终一致性，“用户感知到的一致性”的时间窗口则取决于数据复制到一致状态的时间。基于这些分布式存储系统的特点，本发明提出一种分布式存储QoS的实现方法，请参照图1、2、3。

[0045] 本实施例中的分布式存储QoS的实现方法，分为以下步骤：

[0046] S01、分布式存储运行性能体系建立及获取；

[0047] 在一般分布式存储系统中，主要输出并记录了各进程和存储对象的状态，用于写入或发生故障时触发恢复操作。但没有记录或统计正常运行情况下运行的性能情况，比如对象读写操作次数、失败次数、文件读写频率。

[0048] 本发明中通过在分布式存储软件中增加原子单元的读写操作记录，包括标志、时间和耗时长，结合原子单元所属的文件或归属的其他粒度，统计出冷热文件、原子单元操作频次和发生容量，并将这些指标应用到步骤S02中。

[0049] S02、分布式存储QOS实现；

[0050] 包括：

[0051] S021、排定各待处理文件的优先级；

[0052] 对输入的相关信息计算获得一个优先级数字，所述数字与后续调用一起传递给最终执行程序。

[0053] S022、按照优先级执行顺序操作；

[0054] 排队队列的待处理任务进行优先级从大到小排序，若优先级一样则任务提交时间久的优先。

[0055] 在步骤S021前，首先，将文件按照系统缺省设置的读写块大小拆分为等大小的数据块，最后一个可以为不是等大小的数据块；

[0056] 然后，通过哈希算法获得每个数据块写入磁盘的位置；

[0057] 最后，在磁盘上执行写入或读取。

[0058] 从用户发起读写请求到完成这个请求，主要过程是数据传输、分拆为对象写入磁盘介质几个环节。基于策略的分布式存储QoS可以在两个应用场景中发挥作用：

[0059] A)用户正常应用使用发起的读写请求；

[0060] B)有磁盘损坏替换新盘后数据同步；

[0061] 无论哪种应用场景，都是通过QoS调控，让不同的用户感受到不同的性能，提升用户的满意度和使用体验。

[0062] 实现QoS的过程首先要确认优先级的权值，然后是执行过程中的使用。从分布式存储端到端来看传输的QoS由网络设备根据计算产生的访问权值实现流量调控，分布式存储系统内部的调控由本发明实现。

[0063] QoS权值考虑主观和客观两类指标组成，并通过加权计算最后获得加权值。本发明用来计算QoS权值的指标体系如下表所示：

[0064]

指标名称	指标含义	加权	来源
客户等级	客户的等级级别，最高100，最低0	40%	CRM评价
应用等级	应用重要程度，最高100，最低0	25%	CRM及应用评价
SLA	SLA值危险性得分，最高100	20%	由统计指标提供
文件冷热度	100最热，0最冷	15%	由统计指标提供

[0065] 注：第一次处理的新文件热度值为100。

[0066] 计算完成后的加权值与文件一起在整个请求期间传递使用，发生拥塞时自动按照优先级权值排序，权值越高越优先执行。

[0067] QoS调控在故障恢复时体现的更明显，一块3T的SATA磁盘替换后的数据同步大概需要3-4个小时。在这个过程中，用户使用存储的数据会受到影响。找出重要客户、重要应用和热数据优先恢复。在总时间不能变化的前提下，不同的用户有不同的等待时间是合理的方法。其过程是正常访问的逆操作，即根据当前磁盘上存储的数据信息，从map关系里获取数据归属，根据上层的归属确定哪些数据具有高优先级，哪些数据是低优先级。根据优先级顺序，依次恢复数据。

[0068] S03、分布式存储CAP均衡实现：

[0069] 按照CAP理论，分区容错是分布式存储的前提和基础，是必须实现的要素之一，剩下的一致性和可用性只能选择其一。所谓可用性就是用户体验感知的存储运行性能，而一

致性是同一个数据几个副本之间的实时一致。一般分布式存储从自身可用性和可维护性角度考虑，都优先实现了一致性，而放弃了用户的可用性感受。假设数据dBlock要写入从Node-1到Node-n(Node为副本)的几个存储位置，写入请求为req，完成标志为ack，执行时长为time。

[0070] 则强一致性表现为:ReqNode-1 {ReqNode-2|ReqNode-n}{AckNode-2|AckNode-n} AckNode-1，则用户等待的时间是所有操作完成的时间，即sum(timem|m＝1..n)，此时用户感知的可用性较差，仅实现数据之间的强一致性。数据强一致性是为了在发生故障时，能够无损实现数据同步。

[0071] 虽然在云中心大样本的前提下，硬盘损坏是必然发生的事件，但对于一块具体的硬盘来说损坏不是必然事件，并且都会有较长的正常使用时间。基于这个假设，可以考虑仅业务直接操作的主本完成即算本次操作完成，其他节点的操作可以异步实现，为了保证数据的一致性，增加中间状态变量Flag，其过程表现为：

[0072] ReqNode-1 AckNode-1{FlagNode-2，,FlagNode-n}，Flag由系统记录并管理，在对应的位置完成后给系统发送AckNode-n通知，则设置FlagNode-n 为FinishNode-n。系统另外设置进程检查，当前队列里Flag的情况对于长时间未完成标志更新的判断其执行情况，如果数据操作进程活动，则提高其执行优先级，如果进程已经退出，则重新发起同步操作。通过系统自查的方式，可以保证数据一致性在错时模式下确保实现，不影响后续的其他操作。在这种模式下，用户的等待时间为TimeNode-1使用感受有明显的提升。

[0073] 为了尽量缩短从Flag到Finish状态的更新时间，采取多种提升性能的手段，比如增加SSD磁盘作为cache提升读写效率;通过采用软件RDMA和网络设备的无损传输(PFC)减轻存储服务器的CPU负荷，加快数据在不同节点间的传输。

[0074] 在副本异步写入操作过程中，有数据块写入操作发生时，判断队列中是否写入操作完成，若完成，根据缓存中待写入副本的数据块形成标志队列，形成队列中数据块对应标志及操作排序的优先级权重，副本写入操作完成，更新队列操作日志，并删除队列中排队的数据块;

[0075] 若未完成，则修改优先级的权重，形成队列中数据块对应标志及操作排序的优先级权重，将副本写入操作完成，更新队列操作日志，并删除队列中排队的数据块。

[0076] 当副本写入队列维护巡检时，若存在操作异常及等待时间长的数据块，则修改优先级权重，形成队列中数据块对应标志及操作排序的优先级权重，副本写入操作完成，更新队列操作日志，并删除队列中排队的数据块。

[0077] 上述具体的实施方式仅是本发明具体的个案，本发明的专利保护范围包括但不限于上述具体的实施方式，任何符合本发明的一种分布式存储QoS的实现方法权利要求书的且任何所述技术领域普通技术人员对其做出的适当变化或者替换，皆应落入本发明的专利保护范围。

[0078] 尽管已经示出和描述了本发明的实施例，对于本领域的普通技术人员而言，可以理解在不脱离本发明的原理和精神的情况下可以对这些实施例进行多种变化、修改、替换和变型，本发明的范围由所附权利要求及其等同物限定。

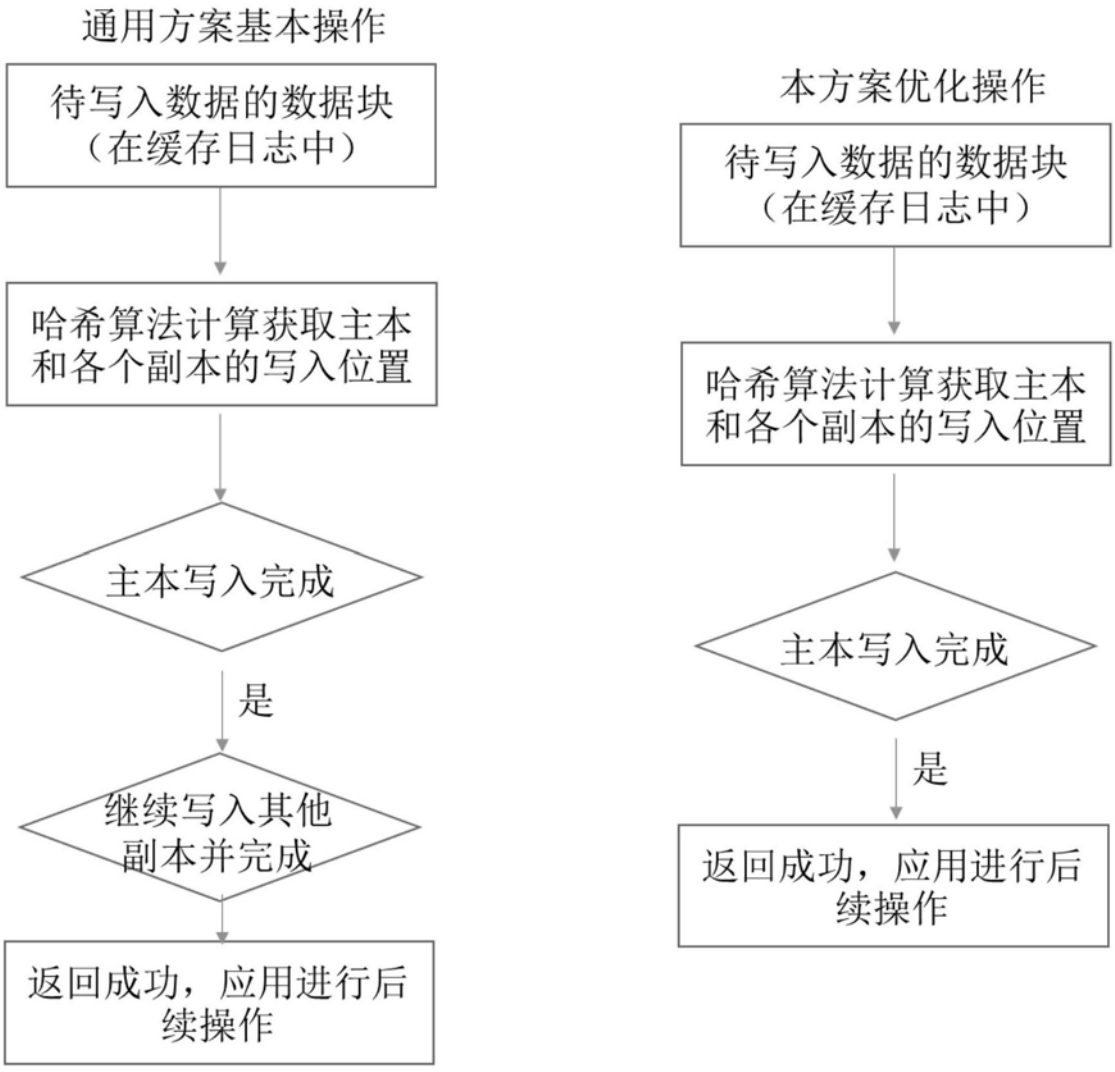

通用方案基本操作
待写入数据的数据块（在缓存日志中）
哈希算法计算获取主本和各个副本的写入位置
主本写入完成
是
继续写入其他副本并完成
返回成功，应用进行后续操作
本方案优化操作
待写入数据的数据块（在缓存日志中）
哈希算法计算获取主本和各个副本的写入位置
主本写入完成
是
返回成功，应用进行后续操作

图1

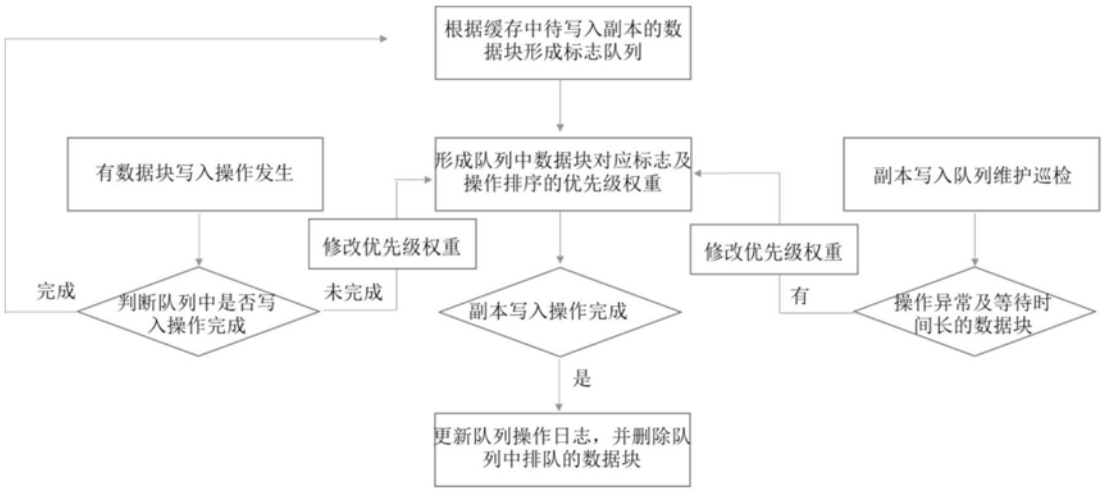

根据缓存中待写入副本的数据块形成标志队列
有数据块写入操作发生
形成队列中数据块对应标志及操作排序的优先级权重
副本写入队列维护巡检
修改优先级权重
修改优先级权重
完成
判断队列中是否写入操作完成
未完成
副本写入操作完成
有
操作异常及等待时间长的数据块
是
更新队列操作日志，并删除队列中排队的数据块

图2

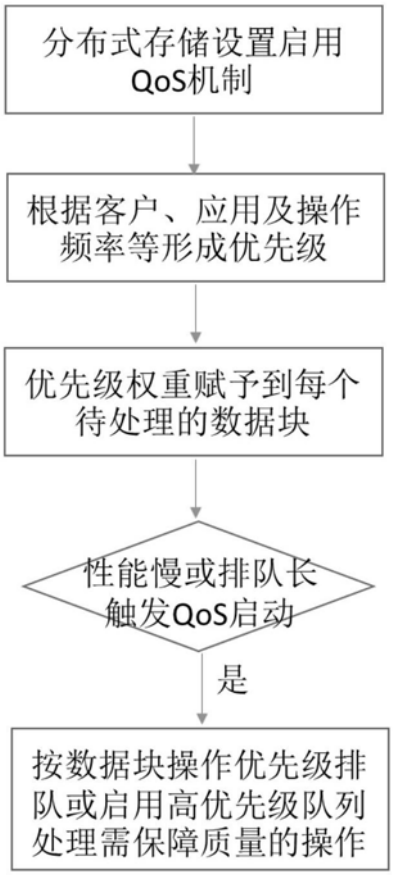

分布式存储设置启用QoS机制
根据客户、应用及操作频率等形成优先级
优先级权重赋予到每个待处理的数据块
性能慢或排队长触发QoS启动
是
按数据块操作优先级排队或启用高优先级队列处理需保障质量的操作

图3