(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 104077423 A
(43)申请公布日 2014.10.01

(21)申请号 201410353123.3

(22)申请日 2014.07.23

(71)申请人 山东大学(威海)
地址 264209 山东省威海市文化西路 180 号

(72)发明人 程杰 杨萌萌

(74)专利代理机构 济南圣达知识产权代理有限公司 37221
代理人 张勇

(51)Int.Cl.
G06F 17/30(2006.01)

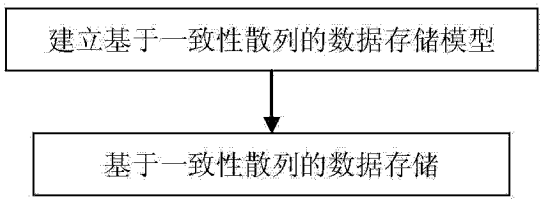
权利要求书3页 说明书9页 附图5页

(54)发明名称

一种基于一致性散列的结构化数据存储、查询和迁移方法

(57)摘要

本发明公开了一种基于一致性散列的结构化数据存储、查询和迁移方法，步骤如下：建立基于一致性散列的 HDFS 数据存储模型，基于此模型进行数据存储和数据查询，当有数据节点加入或失效时，实施数据迁移；所述数据存储方法是将待写入文件的各数据块进行一致性散列得到数据块 Hash 值，然后根据数据块 Hash 值，在节点 Hash 链中查找该数据块的存储节点并将数据块内容存入其存储节点。本发明基于 HDFS 集群主从结构，应用一致性散列，使结构化数据均匀分散在 HDFS 集群的各个数据节点上，有效地提高并行遍历数据的效率，当数据节点数量发生变化时，可大大减少数据迁移所涉及的节点数量和总迁移数据量，提高数据存储系统的运行性能。



CN 104077423 A

1. 基于一致性散列的结构化数据存储方法，其特征是，包括如下步骤：

步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值；对于每一个数据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点。

2. 基于一致性散列的结构化数据存储、查询方法，其特征是，包括如下步骤：

步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值；对于每一个数据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点；

步骤 (3a)：数据查询：首先从名字节点中查找待查询文件所对应的数据块，并计算这些数据块的 Hash 值，然后分别根据所得数据块 Hash 值，按照步骤 (2) 所述的查找方法，在 Hash 链元数据表中查找各数据块所对应的存储节点，在存储节点上进行数据块的读取。

3. 基于一致性散列的结构化数据存储、迁移方法，其特征是，包括如下步骤：

步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值；对于每一个数据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该

数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点；

步骤 (3b)：数据迁移，包括：

步骤 (3b-1)：当部署 HDFS 的集群中有新数据节点加入时，首先计算出新数据节点的 Hash 值，并依据所得 Hash 值，通过二分插入排序算法在 Hash 链元数据表中插入新数据节点的记录，然后将 Hash 链中新数据节点的后继节点上 Hash 值小于或等于新节点 Hash 值的数据块迁移到新节点上，最后在名字节点上对新数据节点及其后继节点的信息进行更新；

步骤 (3b-2)：当部署 HDFS 的集群中出现失效节点时，首先从名字节点中读取该失效节点的信息，计算该失效节点的 Hash 值，并通过二分查找算法在 Hash 链元数据表中找到失效节点的记录，然后将失效节点的数据块从其冗余节点上恢复到失效节点的首个非失效后继节点上，最后从 Hash 链元数据表中删除失效节点记录，从名字节点中删除失效节点信息，并更新恢复节点信息。

4. 如权利要求 1 或 2 或 3 所述的方法，其特征是，所述步骤 (1) 包括：

步骤 (1-1)：计算数据节点 Hash 值：选取一致性 Hash 函数，对部署 HDFS 系统的集群的各数据节点，将其物理地址以 ASCII 码字符串形式作为关键字进行一致性散列，得到各数据节点的 Hash 值；

步骤 (1-2)：构造节点 Hash 链：对于部署 HDFS 系统的集群，将集群中所有数据节点均按照步骤 (1-1) 所述方法计算节点 Hash 值，并根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链；

步骤 (1-3)：存储 Hash 链元数据表：将节点 Hash 链中所有数据节点的物理地址和 Hash 值的映射记录，以顺序表形式存储于 HDFS 系统的名字节点上，形成 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中。

5. 如权利要求 1 或 2 或 3 所述的方法，其特征是，所述步骤 (2) 包括：

将待写入文件所对应的每一个数据块，按照以下步骤进行数据存储，直至所有的数据块均被存储到 HDFS 系统的数据节点中：

步骤 (2-1)：计算数据块 Hash 值：选取与所述步骤 (1-1) 相同的一致性 Hash 函数，以当前数据块的块标号为关键字进行一致性散列，得到当前数据块的 Hash 值；所述数据块标号是指数据块的唯一性标识号；

步骤 (2-2)：查找数据块的存储节点：以当前数据块 Hash 值为查找关键字，通过二分查找算法，在 Hash 链元数据表中查找第一个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所得数据节点即为当前数据块所对应的存储节点；

步骤 (2-3)：存储数据块：将当前数据块内容存储到步骤 (2-2) 查找所得的存储节点上；

步骤 (2-4)：将数据块及其存储节点信息写入名字节点。

6. 如权利要求 2 所述的方法，其特征是，所述步骤 (3a) 包括：

当客户向 HDFS 系统提出读取文件请求时，按以下步骤完成查询：

步骤 (3a-1)：从名字节点中查找该文件所对应的数据块；

步骤 (3a-2)：对每一个数据块分别按照步骤 (2-1) 所述方法计算数据块 Hash 值；

步骤 (3a-3) :按照步骤 (2-2) 所述方法查找当前数据块所对应的存储节点 ;

步骤 (3a-4) :将当前数据块内容从其所对应的存储节点上读出。

7. 如权利要求 3 所述的方法,其特征是,所述步骤 (3b-1) 包括如下步骤 :

步骤 (3b-1-1) :对新加入的数据节点在名字节点进行注册,加入 HDFS 集群中 ;

步骤 (3b-1-2) :按照步骤 (1-1) 所述方法计算新数据节点的 Hash 值 ;

步骤 (3b-1-3) :采用二分插入排序算法,在 Hash 链元数据表中插入新数据节点的物理地址与 Hash 值的映射记录 ;

步骤 (3b-1-4) :在 Hash 链中找到新数据节点的后继节点,将所述后继结点中数据块 Hash 值小于或等于新数据节点 Hash 值的数据块全部迁移到新数据节点上 ;

步骤 (3b-1-5) :在名字节点上对新数据节点及其后继节点的信息进行更新。

8. 如权利要求 3 所述的方法,其特征是,所述步骤 (3b-2) 包括如下步骤 :

步骤 (3b-2-1) :从名字节点中读取失效节点的物理地址、失效节点上各数据块标号及其冗余节点位置 ;

步骤 (3b-2-2) :按照步骤 (1-1) 所述方法计算失效节点的 Hash 值 ;

步骤 (3b-2-3) :根据失效节点的 Hash 值,通过二分查找算法,在 Hash 链元数据表中找到该失效节点,记录其第一个未失效后继节点,作为该失效节点的恢复节点 ;

步骤 (3b-2-4) :对存储于失效数据节点上的所有数据块,将其存储在冗余节点的副本拷贝到步骤 (3b-2-3) 所述的恢复节点上 ;

步骤 (3b-2-5) :从 Hash 链元数据表中删除该失效节点的记录 ;

步骤 (3b-2-6) :从名字节点中删除失效节点信息,并更新恢复节点信息。

一种基于一致性散列的结构化数据存储、查询和迁移方法

技术领域

[0001] 本发明涉及计算机应用技术领域，尤其涉及一种基于一致性散列的结构化数据存储、查询和迁移方法。

背景技术

[0002] 对于海量结构化数据的存储与管理，以 Hadoop 分布式文件系统 (Hadoop Distributed File System,HDFS) 作为底层存储的关系型数据库是目前主要的解决方案。HDFS 的基本思想是将一个文件分成若干个固定大小的数据块进行存储，其架构采用主 / 从结构体系，一个 HDFS 集群包含一个名字节点 (Namenode) 和若干个数据节点 (Datanode)。其中名字节点为主节点，负责控制外部客户机的访问以及存储整个系统的元数据，元数据包括命名空间、文件到数据块的映射、系统配置信息等；而数据节点则是从属节点，用来存储实际文件数据，即 HDFS 数据块。为提高数据的可靠性和可用性，每个数据块都默认保存三份冗余，每个备份副本存储于不同的数据节点上。对外部应用而言，HDFS 如同传统的分布式文件系统，可以对文件进行创建、删除、移动等操作。

[0003] 然而，上述解决方案存在的问题是：

[0004] 1. 存储不均衡，严重影响并行遍历效率

[0005] HDFS 在对表数据进行存储时，是根据集群中各数据节点的负载情况来选择数据块的存储节点的，负载较少的数据节点被优先选择用来存储，这种存储策略不考虑所存储数据块之间的关联，当数据流量很大时，由于大部分数据块会存储到负载较少的节点上，因而使隶属于同一个表的数据块分布不均衡，因而严重降低了遍历数据的并行效率，造成较大的数据库查询延迟。

[0006] 2. 数据迁移涉及所有节点，严重影响系统的运行性能

[0007] 在部署 HDFS 的集群中，数据节点的加入和失效是常态。为保证集群中各个数据节点的负载均衡，当数据节点数量变化时需要进行数据迁移。如：当部署 HDFS 的集群中有新的数据节点加入时，其他所有的节点都要将部分数据迁移到新节点；而当有节点失效时，系统会将失效节点在冗余节点上的备份均匀地迁移到其他节点上。无论是节点加入还是节点失效，数据迁移均涉及 HDFS 系统中所有的数据节点，造成大量的迁移负载，致使网络拥塞，极大地影响了 HDFS 系统的运行性能。

[0008] 一致性散列算法具有以下三个特点：1. 平衡性，即：将关键字进行一致性散列后，可以根据散列值将其均匀地分布在地址空间中。2. 单调性，指当地址空间增大或减小时，通过一致性散列得到的散列值能够映射到新的地址空间，而不是原来的地址空间。3. 分散性，是指当用户通过散列过程将关键字映射到地址空间时，一致性散列算法会避免因可见范围不同而出现的映射结果不一致的情况。一致性散列目前主要用于 P2P 环境和分布式缓存等技术，本发明将一致性散列思想用于 HDFS 结构化数据存储领域，并对现有一致性散列的对等结构进行改进，使其应用于 HDFS 的主从结构体系。

发明内容

[0009] 本发明的目的是为了解决 HDFS 系统所存在的上述两个问题，提供一种基于一致性散列的结构化数据存储、查询和迁移方法，它的优点是：(1) 利用一致性散列对数据块进行存储，使文件所对应的数据块均匀分散在集群中的各个节点上，因而有效地提高了并行遍历数据的效率。(2) 当数据节点数量发生变化时，如：节点增加或失效，只需要在新增节点或失效节点的相邻节点发生数据迁移，因而大大减少数据迁移所涉及的节点数量和总迁移数据量，提高了 HDFS 系统的运行效率。

[0010] 本发明所采用的技术方案如下：

[0011] 定义 1 数据块 Hash 值：对 HDFS 系统中数据块 B，以其数据块标号为关键字进行一致性散列，所得散列值 H_b(B) 称为数据块 B 的 Hash 值。

[0012] 定义 2 节点 Hash 值：对 HDFS 系统中数据节点 D，以其物理地址为关键字进行一致性散列，所得散列值 H_d(D) 称为数据节点 D 的 Hash 值。

[0013] 定义 3 节点 Hash 链：设 $<H\_d_1, H\_d_2, \cdots, H\_d_n>$ 为 HDFS 系统中各数据节点的 Hash 值按照自小到大的顺序进行排序所得序列，其中：$H\_d_k < H\_d_{k+1}$，$(1 \leqslant k < n)$，记 $DN(H\_d_k)$ 表示 H_d_k 所对应的数据节点，则线性结构 $[DN(H\_d_1), DN(H\_d_2), \cdots, DN(H\_d_n)]$ 称为该 HDFS 系统的节点 Hash 链，其中，$DN(H\_d_{k+1})$ 称为 $DN(H\_d_k)$ 的后继节点，同时定义 $DN(H\_d_n)$ 的后继节点为 $DN(H\_d_1)$.

[0014] 基于一致性散列的结构化数据存储方法，包括如下步骤：

[0015] 步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

[0016] 步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值，对于每一个数据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点；

[0017] 基于一致性散列的结构化数据存储、查询方法，包括如下步骤：

[0018] 步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

[0019] 步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值；对于每一个数

据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点；

[0020] 步骤 (3a)：数据查询：首先从名字节点中查找待查询文件所对应的数据块，并计算这些数据块的 Hash 值，然后分别根据所得数据块 Hash 值，按照步骤 (2) 所述的查找方法，在 Hash 链元数据表中查找各数据块所对应的存储节点，在存储节点上进行数据块的读取。

[0021] 基于一致性散列的结构化数据存储、迁移方法，包括如下步骤：

[0022] 步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

[0023] 步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值；对于每一个数据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点；

[0024] 步骤 (3b)：数据迁移，包括：

[0025] 步骤 (3b-1)：当部署 HDFS 的集群中有新数据节点加入时，首先计算出新数据节点的 Hash 值，并依据所得 Hash 值，通过二分插入排序算法在 Hash 链元数据表中插入新数据节点的记录，然后将 Hash 链中新数据节点的后继节点上 Hash 值小于或等于新节点 Hash 值的数据块迁移到新节点上，最后在名字节点上对新数据节点及其后继节点的信息进行更新；

[0026] 步骤 (3b-2)：当部署 HDFS 的集群中出现失效节点时，首先从名字节点中读取该失效节点的信息，计算该失效节点的 Hash 值，并通过二分查找算法在 Hash 链元数据表中找到失效节点的记录，然后将失效节点的数据块从其冗余节点上恢复到失效节点的首个非失效后继节点上，最后从 Hash 链元数据表中删除失效节点记录，从名字节点中删除失效节点信息，并更新恢复节点信息。

[0027] 所述步骤 (1) 包括：

[0028] 步骤 (1-1)：计算数据节点 Hash 值：选取一致性 Hash 函数，对部署 HDFS 系统的集群的各数据节点，将其物理地址以 ASCII 码字符串形式作为关键字进行一致性散列，得到各数据节点的 Hash 值；

[0029] 步骤 (1-2)：构造节点 Hash 链：对于部署 HDFS 系统的集群，将集群中所有数据节点均按照步骤 (1-1) 所述方法计算节点 Hash 值，并根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链；

[0030] 步骤 (1-3) :存储 Hash 链元数据表 :将节点 Hash 链中所有数据节点的物理地址和 Hash 值的映射记录,以顺序表形式存储于 HDFS 系统的名字节点上,形成 Hash 链元数据表,当 HDFS 启动时,所述 Hash 链元数据表将自动加载到名字节点的内存中。

[0031] 所述步骤 (2) 包括 :

[0032] 将待写入文件所对应的每一个数据块,按照以下步骤进行数据存储,直至所有的数据块均被存储到 HDFS 系统的数据节点中 :

[0033] 步骤 (2-1) :计算数据块 Hash 值 :选取与所述步骤 (1-1) 相同的一致性 Hash 函数,以当前数据块的块标号为关键字进行一致性散列,得到当前数据块的 Hash 值 ;所述数据块标号是指数据块的唯一性标识号 ;

[0034] 步骤 (2-2) :查找数据块的存储节点 :以当前数据块 Hash 值为查找关键字,通过二分查找算法,在 Hash 链元数据表中查找第一个节点 Hash 值大于或等于该数据块 Hash 值的数据节点,所得数据节点即为当前数据块所对应的存储节点 ;

[0035] 步骤 (2-3) :存储数据块 :将当前数据块内容存储到步骤 (2-2) 查找所得的存储节点上 ;

[0036] 步骤 (2-4) :将数据块及其存储节点信息写入名字节点。

[0037] 所述步骤 (3a) 包括 :

[0038] 当客户向 HDFS 系统提出读取文件请求时,按以下步骤完成查询 :

[0039] 步骤 (3a-1) :从名字节点中查找该文件所对应的数据块 ;

[0040] 步骤 (3a-2) :对每一个数据块分别按照步骤 (2-1) 所述方法计算数据块 Hash 值 ;

[0041] 步骤 (3a-3) :按照步骤 (2-2) 所述方法查找当前数据块所对应的存储节点 ;

[0042] 步骤 (3a-4) :将当前数据块内容从其所对应的存储节点上读出。

[0043] 所述步骤 (3b-1) 包括如下步骤 :

[0044] 步骤 (3b-1-1) :对新加入的数据节点在名字节点进行注册,加入 HDFS 集群中 ;

[0045] 步骤 (3b-1-2) :按照步骤 (1-1) 所述方法计算新数据节点的 Hash 值 ;

[0046] 步骤 (3b-1-3) :采用二分插入排序算法,在 Hash 链元数据表中插入新数据节点的物理地址与 Hash 值的映射记录 ;

[0047] 步骤 (3b-1-4) :在 Hash 链中找到新数据节点的后继节点,将所述后继结点中数据块 Hash 值小于或等于新数据节点 Hash 值的数据块全部迁移到新数据节点上 ;

[0048] 步骤 (3b-1-5) :在名字节点上对新数据节点及其后继节点的信息进行更新。

[0049] 所述步骤 (3b-2) 包括如下步骤 :

[0050] 步骤 (3b-2-1) :从名字节点中读取失效节点的物理地址、失效节点上各数据块标号及其冗余节点位置 ;

[0051] 步骤 (3b-2-2) :按照步骤 (1-1) 所述方法计算失效节点的 Hash 值 ;

[0052] 步骤 (3b-2-3) :根据失效节点的 Hash 值,通过二分查找算法,在 Hash 链元数据表中找到该失效节点,记录其第一个未失效后继节点,作为该失效节点的恢复节点 ;

[0053] 步骤 (3b-2-4) :对存储于失效数据节点上的所有数据块,将其存储在冗余节点的副本拷贝到步骤 (3b-2-3) 所述的恢复节点上 ;

[0054] 步骤 (3b-2-5) :从 Hash 链元数据表中删除该失效节点的记录 ;

[0055] 步骤 (3b-2-6) :从名字节点中删除失效节点信息,并更新恢复节点信息。

[0056] 本发明的有益效果：

[0057] (1) 本发明基于一致性散列对 HDFS 系统的数据块进行存储，每个数据块根据 Hash 值确定所对应的存储节点，由于一致性散列能够使文件所对应的数据块均匀分散在集群的各个数据节点上，因而大大提高了并行遍历数据的效率。

[0058] (2) 当数据节点数量发生变化时，如：节点加入或失效，只需要在新添节点或失效节点的相邻节点发生数据迁移，大大减少了数据迁移所涉及的节点数量和总迁移数据量，从而有效地提高了 HDFS 系统的运行性能。

附图说明

[0059] 图 1 是本发明的基于一致性散列的结构化数据存储主流程图；

[0060] 图 2 是本发明的基于一致性散列的结构化数据存储、查询主流程图；

[0061] 图 3 是本发明的基于一致性散列的结构化数据存储、迁移主流程图；

[0062] 图 4 是数据节点 Hash 链结构示意图；

[0063] 图 5 是节点 Hash 链元数据表示意图；

[0064] 图 6 是数据节点 Hash 链构造过程示意图；

[0065] 图 7 是 HDFS 数据块存储过程示意图；

[0066] 图 8 是节点添加时数据迁移过程示意图；

[0067] 图 9 是节点失效时数据迁移过程示意图。

具体实施方式

[0068] 下面结合附图与实施例对本发明作进一步说明。

[0069] 定义 1 数据块 Hash 值：对 HDFS 系统中数据块 B，以其数据块标号为关键字进行一致性散列，所得散列值 H_b(B) 称为数据块 B 的 Hash 值。

[0070] 定义 2 节点 Hash 值：对 HDFS 系统中数据节点 D，以其物理地址为关键字进行一致性散列，所得散列值 H_d(D) 称为数据节点 D 的 Hash 值。

[0071] 定义 3 节点 Hash 链：设 $<H\_d_1, H\_d_2, \cdots, H\_d_n>$ 为 HDFS 系统中各数据节点的 Hash 值按照自小到大的顺序进行排序所得序列，其中：$H\_d_k < H\_d_{k+1}$，$(1 \leqslant k < n)$，记 $DN(H\_d_k)$ 表示 H_d_k 所对应的数据节点，则线性结构 $[DN(H\_d_1), DN(H\_d_2), \cdots, DN(H\_d_n)]$ 称为该 HDFS 系统的节点 Hash 链，其中，$DN(H\_d_{k+1})$ 称为 $DN(H\_d_k)$ 的后继节点，同时定义 $DN(H\_d_n)$ 的后继节点为 $DN(H\_d_1)$.

[0072] 如图 1 所示，基于一致性散列的结构化数据存储方法，包括如下步骤：

[0073] 步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

[0074] 步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值；对于每一个数

据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点；

[0075] 如图 2 所示，基于一致性散列的结构化数据存储、查询方法，包括如下步骤：

[0076] 步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

[0077] 步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值；对于每一个数据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点；

[0078] 步骤 (3a)：数据查询：首先从名字节点中查找待查询文件所对应的数据块，并计算这些数据块的 Hash 值，然后分别根据所得数据块 Hash 值，按照步骤 (2) 所述的查找方法，在 Hash 链元数据表中查找各数据块所对应的存储节点，在存储节点上进行数据块的读取。

[0079] 如图 3 所示，基于一致性散列的结构化数据存储、迁移方法，包括如下步骤：

[0080] 步骤 (1)：建立数据存储模型：首先对部署 HDFS 的集群中所有数据节点，以数据节点的物理地址为关键字进行一致性散列，得到节点 Hash 值；然后根据所述节点 Hash 值由小到大对数据节点进行排序，形成节点 Hash 链，将节点 Hash 链中所有数据节点的物理地址与 Hash 值的映射记录以顺序表形式存储到 HDFS 集群的名字节点上，所述映射记录顺序表又称 Hash 链元数据表，当 HDFS 启动时，所述 Hash 链元数据表将自动加载到名字节点的内存中；

[0081] 步骤 (2)：数据存储：将待写入文件的每个数据块按照数据块标号，采用与所述步骤 (1) 数据节点散列相同的哈希函数进行一致性散列，得到数据块 Hash 值；对于每一个数据块，首先根据其数据块 Hash 值，从 Hash 链元数据表中查找首个节点 Hash 值大于或等于该数据块 Hash 值的数据节点，所查找的数据节点即为该数据块所对应的存储节点，然后将当前数据块内容存储到所对应的存储节点上，最后将数据块及其存储节点的信息写入名字节点；

[0082] 步骤 (3b)：数据迁移，包括：

[0083] 步骤 (3b-1)：当部署 HDFS 的集群中有新数据节点加入时，首先计算出新数据节点的 Hash 值，并依据所得 Hash 值，通过二分插入排序算法在 Hash 链元数据表中插入新数据节点的记录，然后将 Hash 链中新数据节点的后继节点上 Hash 值小于或等于新节点 Hash 值的数据块迁移到新节点上，最后在名字节点上对新数据节点及其后继节点的信息进行更

新；

[0084] 步骤（3b-2）：当部署HDFS的集群中出现失效节点时，首先从名字节点中读取该失效节点的信息，计算该失效节点的Hash值，并通过二分查找算法在Hash链元数据表中找到失效节点的记录，然后将失效节点的数据块从其冗余节点上恢复到失效节点的首个非失效后继节点上，最后从Hash链元数据表中删除失效节点记录，从名字节点中删除失效节点信息，并更新恢复节点信息。

[0085] 所述步骤（1）包括：

[0086] 步骤（1-1）：计算数据节点Hash值：选取一致性Hash函数，对部署HDFS系统的集群的各数据节点，将其物理地址以ASCII码字符串形式作为关键字进行一致性散列，得到各数据节点的Hash值；

[0087] 步骤（1-2）：构造节点Hash链：对于部署HDFS系统的集群，将集群中所有数据节点均按照步骤（1-1）所述方法计算节点Hash值，并根据所述节点Hash值由小到大对数据节点进行排序，形成节点Hash链；

[0088] 步骤（1-3）：存储Hash链元数据表：将节点Hash链中所有数据节点的物理地址和Hash值的映射记录，以顺序表形式存储于HDFS系统的名字节点上，形成Hash链元数据表，当HDFS启动时，所述Hash链元数据表将自动加载到名字节点的内存中。

[0089] 所述步骤（2）包括：

[0090] 将待写入文件所对应的每一个数据块，按照以下步骤进行数据存储，直至所有的数据块均被存储到HDFS系统的数据节点中：

[0091] 步骤（2-1）：计算数据块Hash值：选取与所述步骤（1-1）相同的一致性Hash函数，以当前数据块的块标号为关键字进行一致性散列，得到当前数据块的Hash值；所述数据块标号是指数据块的唯一性标识号；

[0092] 步骤（2-2）：查找数据块的存储节点：以当前数据块Hash值为查找关键字，通过二分查找算法，在Hash链元数据表中查找第一个节点Hash值大于或等于该数据块Hash值的数据节点，所得数据节点即为当前数据块所对应的存储节点；

[0093] 步骤（2-3）：存储数据块：将当前数据块内容存储到步骤（2-2）查找所得的存储节点上；

[0094] 步骤（2-4）：将数据块及其存储节点信息写入名字节点。

[0095] 所述步骤（3a）包括：

[0096] 当客户向HDFS系统提出读取文件请求时，按以下步骤完成查询：

[0097] 步骤（3a-1）：从名字节点中查找该文件所对应的数据块；

[0098] 步骤（3a-2）：对每一个数据块分别按照步骤（2-1）所述方法计算数据块Hash值；

[0099] 步骤（3a-3）：按照步骤（2-2）所述方法查找当前数据块所对应的存储节点；

[0100] 步骤（3a-4）：将当前数据块内容从其所对应的存储节点上读出。

[0101] 所述步骤（3b-1）包括如下步骤：

[0102] 步骤（3b-1-1）：对新加入的数据节点在名字节点进行注册，加入HDFS集群中；

[0103] 步骤（3b-1-2）：按照步骤（1-1）所述方法计算新数据节点的Hash值；

[0104] 步骤（3b-1-3）：采用二分插入排序算法，在Hash链元数据表中插入新数据节点的物理地址与Hash值的映射记录；

[0105] 步骤 (3b-1-4) :在 Hash 链中找到新数据节点的后继节点,将所述后继结点中数据块 Hash 值小于或等于新数据节点 Hash 值的数据块全部迁移到新数据节点上 ;

[0106] 步骤 (3b-1-5) :在名字节点上对新数据节点及其后继节点的信息进行更新。

[0107] 所述步骤 (3b-2) 包括如下步骤 :

[0108] 步骤 (3b-2-1) :从名字节点中读取失效节点的物理地址、失效节点上各数据块标号及其冗余节点位置 ;

[0109] 步骤 (3b-2-2) :按照步骤 (1-1) 所述方法计算失效节点的 Hash 值 ;

[0110] 步骤 (3b-2-3) :根据失效节点的 Hash 值,通过二分查找算法,在 Hash 链元数据表中找到该失效节点,记录其第一个未失效后继节点,作为该失效节点的恢复节点 ;

[0111] 步骤 (3b-2-4) :对存储于失效数据节点上的所有数据块,将其存储在冗余节点的副本拷贝到步骤 (3b-2-3) 所述的恢复节点上 ;

[0112] 步骤 (3b-2-5) :从 Hash 链元数据表中删除该失效节点的记录 ;

[0113] 步骤 (3b-2-6) :从名字节点中删除失效节点信息,并更新恢复节点信息。

[0114] 如图 4 所示,数据节点 Hash 链结构,以五个数据节点 A ～ E 为例,图中数据节点旁边方框中字符表示该数据节点的 Hash 值,方框下方字符表示该数据节点的物理地址,箭头表示后继关系。

[0115] 如图 5 所示,节点 Hash 链元数据表,图中 $HA(Node_j)$ 表示数据节点 $Node_j$ 的物理地址,$Hash(Node_j)$ 表示数据节点 $Node_j$ 的 Hash 值,其中,$1 \leqslant j \leqslant n$,n 为数据节点数量 ;且对于节点 $Node_k(1 \leqslant k<n)$,$Hash(Node_k) < Hash(Node_{k+1})$.

[0116] 如图 6 所示,数据节点 Hash 链构造过程,包括如下步骤 :

[0117] 步骤 (101) :读取各数据节点的物理地址 ;

[0118] 步骤 (102) :计算各数据节点 Hash 值 ;

[0119] 步骤 (103) :将数据节点按其 Hash 值从小到大的顺序排序 ;

[0120] 步骤 (104) :将数据节点的物理地址与其节点 Hash 值的映射记录依次写入 Hash 链元数据表。

[0121] 如图 7 所示,HDFS 数据块存储过程,包括如下步骤 :

[0122] 步骤 (201) :判断所有数据块是否已经存储,如果是就结束 ;如果否就进入步骤 (202) ;

[0123] 步骤 (202) :读取一个未存储的数据块 ;

[0124] 步骤 (203) :计算当前数据块 Hash 值 ;

[0125] 步骤 (204) :利用二分查找算法在节点 Hash 链中查找当前数据块所对应的存储节点 ;

[0126] 步骤 (205) :将数据块写入此存储节点 ;

[0127] 步骤 (206) :将数据块及其所对应的存储节点信息写入名字节点。

[0128] 如图 8 所示,节点添加时数据迁移的步骤如下 :

[0129] 步骤 (301) :对新加入的节点在名字节点进行注册 ;

[0130] 步骤 (302) :计算新加入数据节点的 Hash 值 ;

[0131] 步骤 (303) :二分遍历节点 Hash 链元数据,找到新节点在 Hash 链中的插入位置 ;

[0132] 步骤 (304) :将新节点的物理地址与 Hash 值的映射记录插入 Hash 链元数据表中 ;

[0133] 步骤（305）：将新节点的后继节点中数据块 Hash 值小于或等于新节点 Hash 值的数据块全部迁移到新节点上；

[0134] 步骤（306）：在名字节点上对新数据节点及其后继节点的信息进行更新。

[0135] 如图 9 所示，节点失效时数据迁移的步骤如下：

[0136] 步骤（401）：从名字节点中读取失效节点信息；

[0137] 步骤（402）：计算失效节点的 Hash 值；

[0138] 步骤（403）：二分查找节点 Hash 链，确定失效节点在节点 Hash 链的位置；

[0139] 步骤（404）：记录该失效节点的第一个未失效后继节点信息；

[0140] 步骤（405）：将失效节点的数据块内容从其冗余节点恢复到失效节点的第一个未失效后继节点上；

[0141] 步骤（406）：从节点 Hash 链元数据表中删除失效节点记录；

[0142] 步骤（407）：从名字节点中删除失效节点信息，并更新恢复节点信息。

[0143] 上述虽然结合附图对本发明的具体实施方式进行了描述，但并非对本发明保护范围的限制，所属领域技术人员应该明白，在本发明的技术方案的基础上，本领域技术人员不需要付出创造性劳动即可做出的各种修改或变形仍在本发明的保护范围以内。

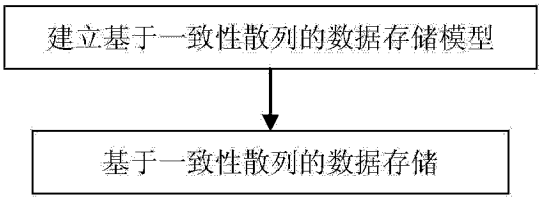

建立基于一致性散列的数据存储模型
基于一致性散列的数据存储

图 1

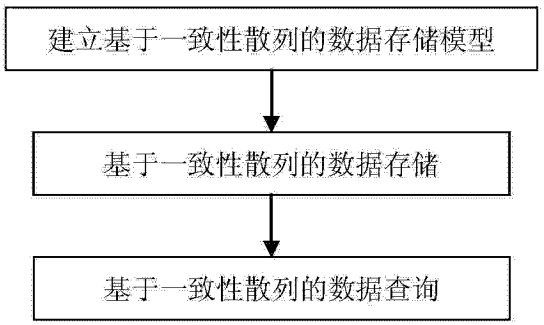

建立基于一致性散列的数据存储模型
基于一致性散列的数据存储
基于一致性散列的数据查询

图 2

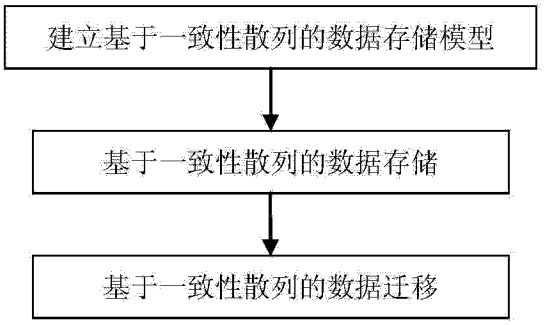

建立基于一致性散列的数据存储模型
基于一致性散列的数据存储
基于一致性散列的数据迁移

图 3

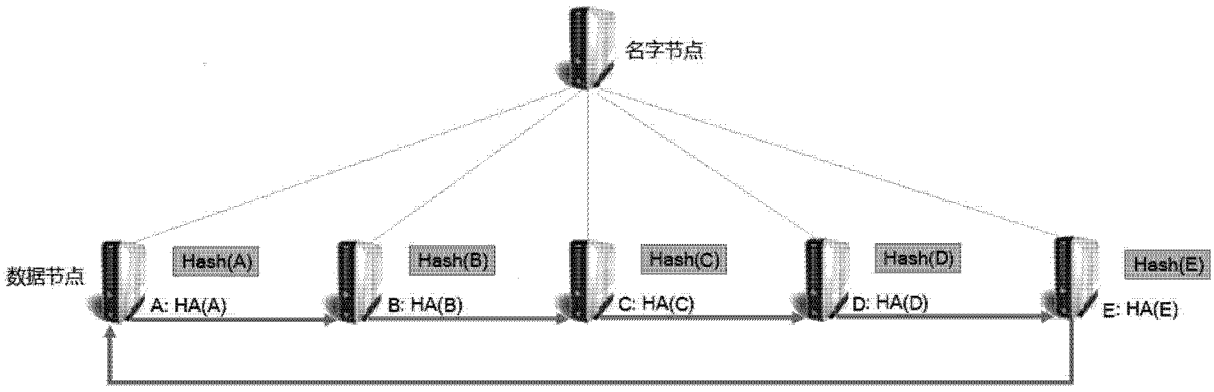

名字节点
数据节点
Hash(A)
A: HA(A)
Hash(B)
B: HA(B)
Hash(C)
C: HA(C)
Hash(D)
D: HA(D)
Hash(E)
E: HA(E)

图 4

$HA(Node_1)$	$Hash(Node_1)$
$HA(Node_2)$	$Hash(Node_2)$
…	…
$HA(Node_n)$	$Hash(Node_n)$

图 5

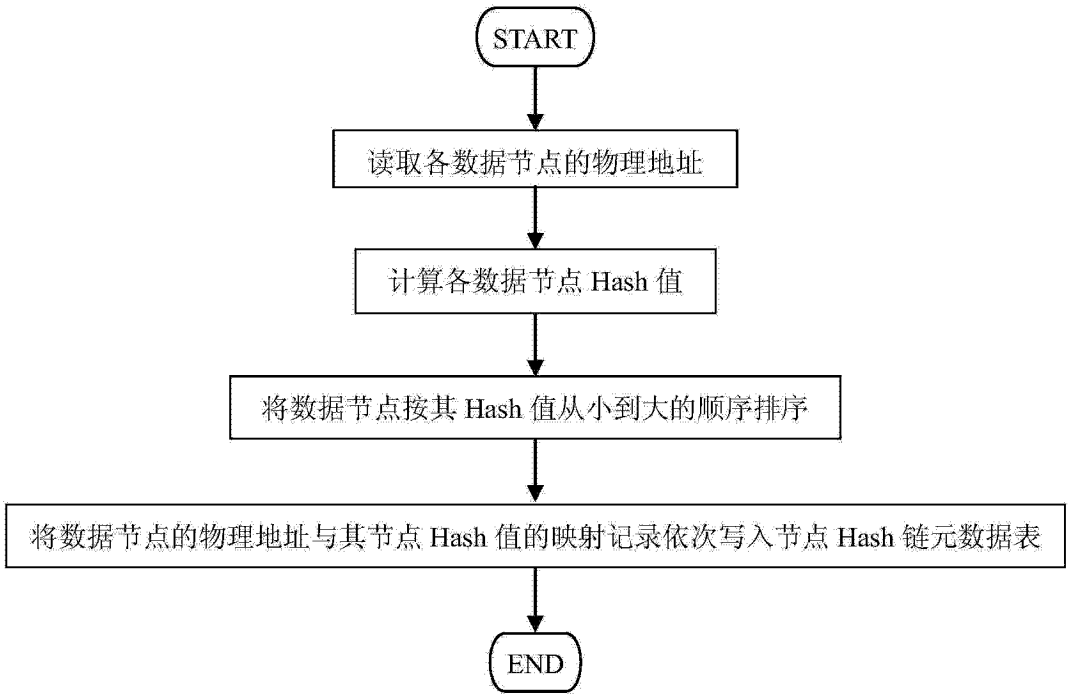


图 6

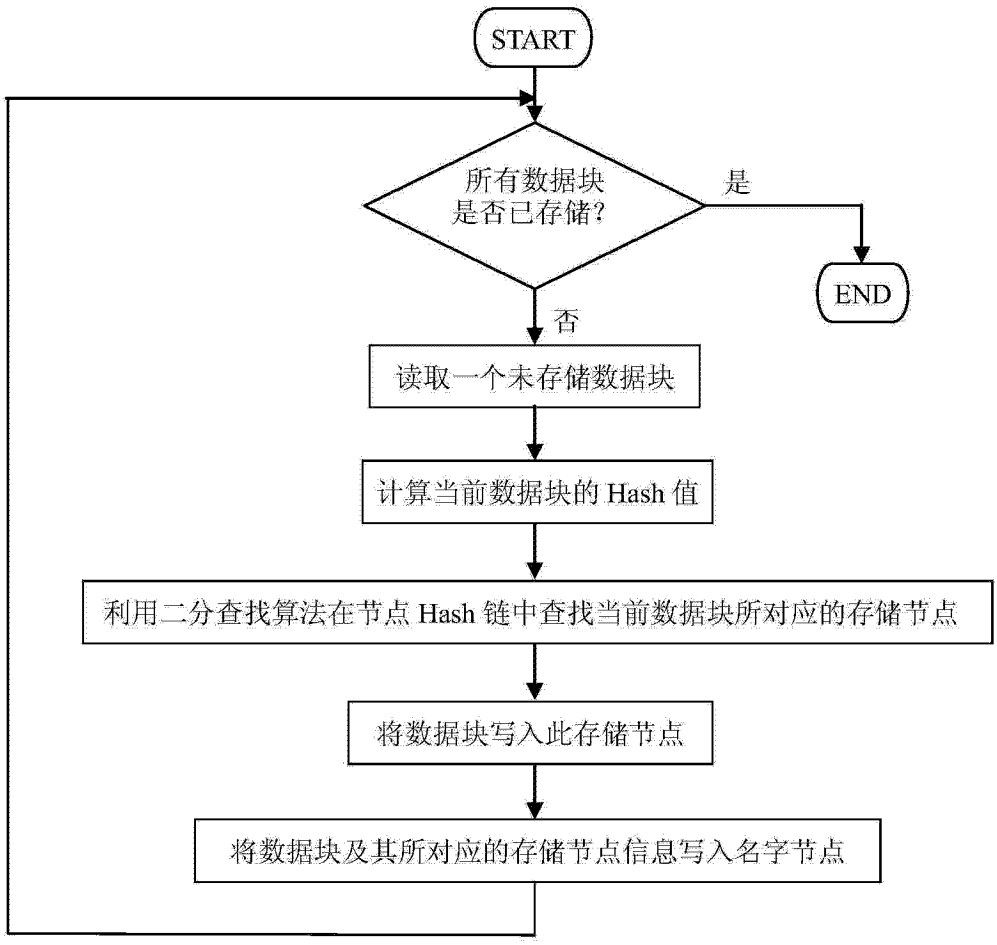

START
所有数据块
是否已存储?
是
END
否
读取一个未存储数据块
计算当前数据块的 Hash 值
利用二分查找算法在节点 Hash 链中查找当前数据块所对应的存储节点
将数据块写入此存储节点
将数据块及其所对应的存储节点信息写入名字节点

图 7

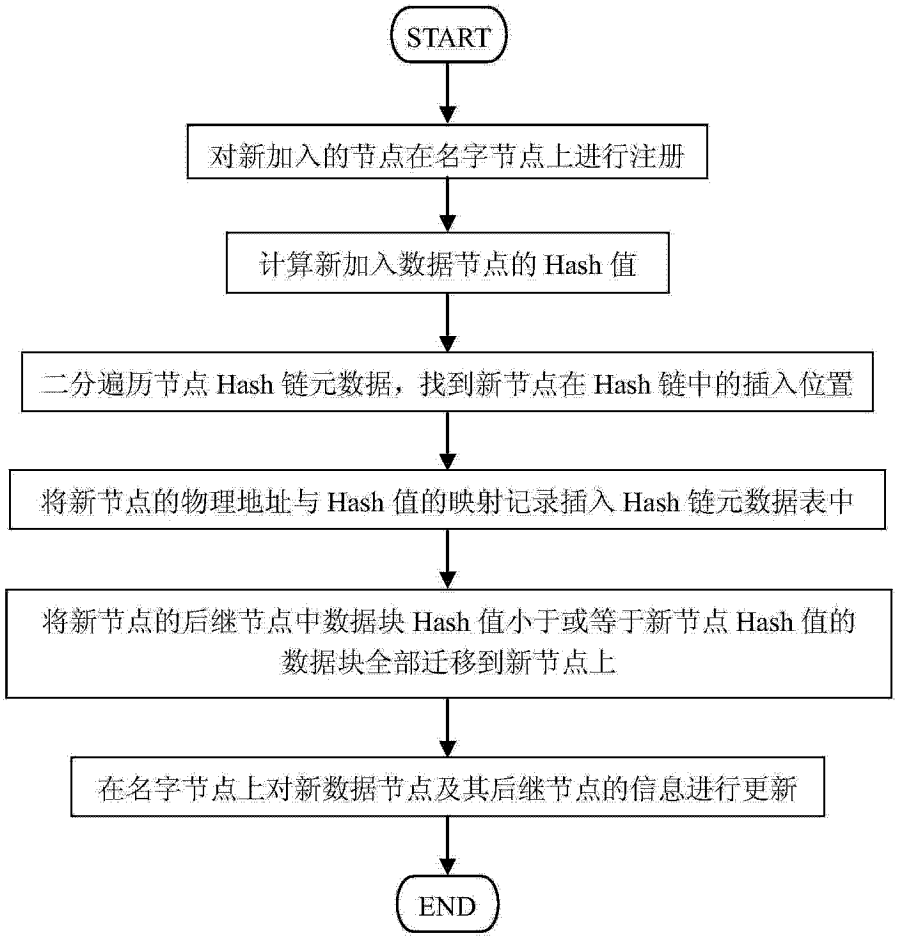

START
对新加入的节点在名字节点上进行注册
计算新加入数据节点的 Hash 值
二分遍历节点 Hash 链元数据，找到新节点在 Hash 链中的插入位置
将新节点的物理地址与 Hash 值的映射记录插入 Hash 链元数据表中
将新节点的后继节点中数据块 Hash 值小于或等于新节点 Hash 值的数据块全部迁移到新节点上
在名字节点上对新数据节点及其后继节点的信息进行更新
END

图 8

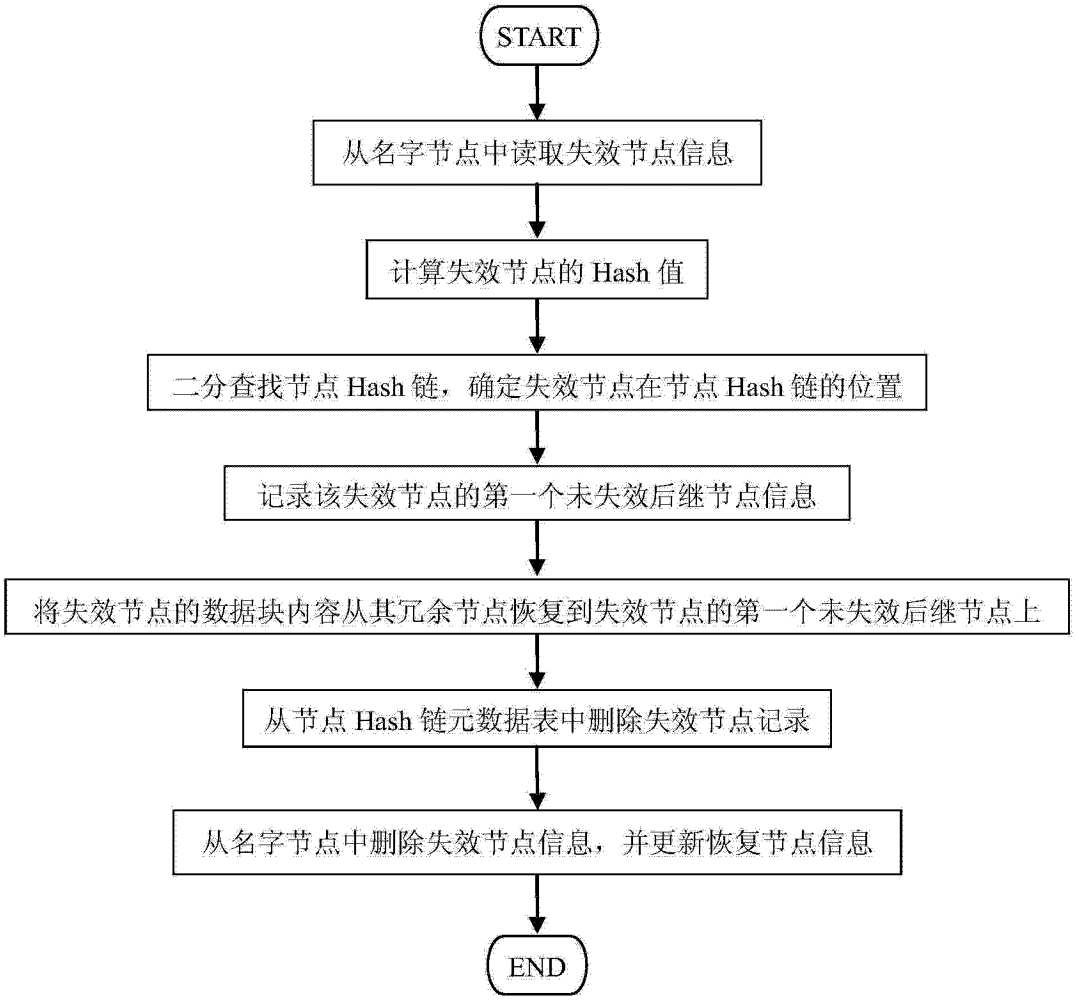

START
从名字节点中读取失效节点信息
计算失效节点的 Hash 值
二分查找节点 Hash 链，确定失效节点在节点 Hash 链的位置
记录该失效节点的第一个未失效后继节点信息
将失效节点的数据块内容从其冗余节点恢复到失效节点的第一个未失效后继节点上
从节点 Hash 链元数据表中删除失效节点记录
从名字节点中删除失效节点信息，并更新恢复节点信息
END

图 9