(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国 际 局

(43) 国际公布日
2020 年 1 月 16 日 (16.01.2020)

WIPO | PCT

(10) 国际公布号
WO 2020/010502 A1

(51) 国际专利分类号:
*H04L 29/08* (2006.01)

(21) 国际申请号: PCT/CN2018/095082

(22) 国际申请日: 2018 年 7 月 10 日 (10.07.2018)

(25) 申请语言: 中文

(26) 公布语言: 中文

(71) 申请人: 深圳花儿数据技术有限公司 (**HERE DATA TECHNOLOGY**) [CN/CN]; 中国广东省深圳市前海深港合作区前海一路 1 号 A 栋 201室, Guangdong 518000 (CN)。

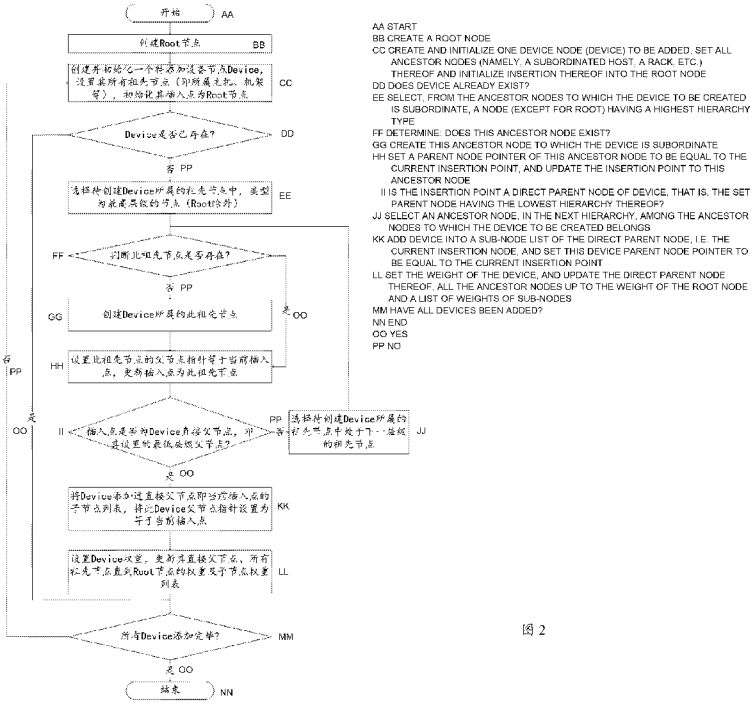
(72) 发明人: 郝斌 (**HAO, Bin**); 中国广东省深圳市前海深港合作区前海一路 1 号 A 栋 201 室, Guangdong 518000 (CN)。

(74) 代理人: 深圳鼎合诚知识产权代理有限公司 (**DHC IP ATTORNEYS**); 中国广东省深圳福田区金田路与福华路交汇处现代国际大厦2201, Guangdong 518048 (CN)。

(81) 指定国(除另有指明，要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW。

(54) **Title:** DISTRIBUTED DATA REDUNDANT STORAGE METHOD BASED ON CONSISTENT HASH ALGORITHM

(54) 发明名称: 一种基于一致性哈希算法的分布式数据冗余存储方法




AA START
BB CREATE A ROOT NODE
CC CREATE AND INITIALIZE ONE DEVICE NODE (DEVICE) TO BE ADDED, SET ALL ANCESTOR NODES (NAMELY, A SUBORDINATED HOST, A RACK, ETC.) THEREOF AND INITIALIZE INSERTION THEREOF INTO THE ROOT NODE
DD DOES DEVICE ALREADY EXIST?
EE SELECT, FROM THE ANCESTOR NODES TO WHICH THE DEVICE TO BE CREATED IS SUBORDINATE, A NODE (EXCEPT FOR ROOT) HAVING A HIGHEST HIERARCHY TYPE
FF DETERMINE: DOES THIS ANCESTOR NODE EXIST?
GG CREATE THIS ANCESTOR NODE TO WHICH THE DEVICE IS SUBORDINATE
HH SET A PARENT NODE POINTER OF THIS ANCESTOR NODE TO BE EQUAL TO THE CURRENT INSERTION POINT, AND UPDATE THE INSERTION POINT TO THIS ANCESTOR NODE
II IS THE INSERTION POINT A DIRECT PARENT NODE OF DEVICE, THAT IS, THE SET PARENT NODE HAVING THE LOWEST HIERARCHY THEREOF?
JJ SELECT AN ANCESTOR NODE, IN THE NEXT HIERARCHY, AMONG THE ANCESTOR NODES TO WHICH THE DEVICE TO BE CREATED BELONGS
KK ADD DEVICE INTO A SUB-NODE LIST OF THE DIRECT PARENT NODE, I.E. THE CURRENT INSERTION NODE, AND SET THIS DEVICE PARENT NODE POINTER TO BE EQUAL TO THE CURRENT INSERTION POINT
LL SET THE WEIGHT OF THE DEVICE, AND UPDATE THE DIRECT PARENT NODE THEREOF, ALL THE ANCESTOR NODES UP TO THE WEIGHT OF THE ROOT NODE AND A LIST OF WEIGHTS OF SUB-NODES
MM HAVE ALL DEVICES BEEN ADDED?
NN END
OO YES
PP NO

图 2

(57) **Abstract:** A distributed data redundant storage method based on a consistent hash algorithm for the selection of a distribution position of pre-stored data in a storage cluster. The method comprises: first establishing a topologic structure of each storage node in a distributed storage system, and determining the position of each node in a storage node sequence corresponding to each hash subspace using the consistent hash algorithm; then determining the number of copies of data and each copy of storage data in a pre-stored data redundant storage manner; and storing each copy of storage data in a different storage node according to a storage rule. By means of

WO 2020/010502 A1

[见续页]

(84) 指定国(除另有指明，要求每一种可提供的地区保护)：ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布：
— 包括国际检索报告(条约第21条(3))。

the method, an organization architecture of a cluster in a distributed storage system is rapidly established, and a supervised weight random-selection algorithm is used to achieve the objective of efficiently selecting a storage node when a target storage node sequence is selected for pre-stored data in a cluster, and therefore, the storage requirement of massive data is satisfied.

(57) 摘要：一种基于一致性哈希算法的分布式数据冗余存储方法，用于预存数据在存储集群中分布位置的选择，该方法先建立分布式存储系统中各存储节点的拓扑逻辑结构，并通过一致性哈希算法，确定各个哈希子空间所对应存储节点序列中各节点位置，然后依据预存储数据冗余存储方式，确定数据份数和每一份存储数据，依据存储规则将所述每一份存储数据存储到不同的存储节点中。本方法快速建立分布式存储系统的集群组织架构，并利用带监督的权重随机选择算法，实现在集群中为预存储数据选择目标存储节点序列时高效选择存储节点的目的，进而满足海量数据的存储要求。

# 一种基于一致性哈希算法的分布式数据冗余存储方法

## 技术领域

本发明涉及分布式存储领域，具体涉及一种基于一致性哈希算法的分布式数据冗余存储方法。

## 背景技术

当前已处于云计算广泛普及、应用繁荣发展的新阶段，数据规模呈爆炸性增长，海量数据安全高效存储成为云计算关键技术。传统存储技术，如 NFS、SAN 等，从可扩展性、容错性、可用性、部署成本等方面，已无法或很难满足海量数据（PB 或 EB 级）的存储要求。

## 发明内容

本申请公开了一种基于一致性哈希算法的分布式数据冗余存储方法，解决背景技术中传统存储技术无法或很难满足海量数据存储要求的问题。

根据第一方面，一种实施例中提供一种基于一致性哈希算法的分布式数据冗余存储方法，该方法包括：

节点创建步骤：建立分布式存储系统中各存储节点的拓扑逻辑结构，并通过一致性哈希算法，确定各个哈希子空间所对应存储节点序列中各节点的位置；

数据写入步骤：依据预存储数据冗余存储方式，确定数据份数和每一份存储数据；

依据存储规则将所述每一份存储数据存储到不同的存储节点中。

依据上述实施例的一种基于一致性哈希算法的分布式数据冗余存储方法，将存储数据存储到不同存储节点中，只计算一次哈希用于将对象标识映射哈希子空间，然后利用带监督的权重随机选择算法选择哈希子空间对应的存储节点列表，提高节点选择效率，进而满足海量数据存储的要求。

## 附图说明

图 1 为基于物理存储设备构建的集群组织架构示意图；

图 2 为根据物理存储设备在集群组织架构中的位置添加节点的流程图；

图 3 为哈希环划分及对象路由过程示意图；

图 4 为节点选择总体流程图；

图 5 为带监督的权重随机节点选择流程图。

## 具体实施方式

下面通过具体实施方式结合附图对本发明作进一步详细说明。其中不同实施方式中类似元件采用了相关联的类似的元件标号。在以下的实施方式中，很多细节描述是为了使得本申请能被更好的理解。然而，本领域技术人员可以毫不费力的认识到，其中部分特征在不同情况下是可以省略的，或者可以由其他元件、材料、方法所替代。在某些情况下，本申请相关的一些操作并没有在说明书中显示或者描述，这是为了避免本申请的核心部分被过多的描述所淹没，而对于本领域技术人员而言，详细描述这些相关操作并不是必要的，他们根据说明书中的描述以及本领域的一般技术知识即可完整了解相关操作。

另外，说明书中所描述的特点、操作或者特征可以以任意适当的方式结合形成各种实施方式。同时，方法描述中的各步骤或者动作也可以按照本领域技术人员所能显而易见的方式进行顺序调换或调整。因此，说明书和附图中的各种顺序只是为了清楚描述某一个实施例，并不意味着是必须的顺序，除非另有说明其中某个顺序是必须遵循的。

本文中为部件所编序号本身，例如“第一”、“第二”等，仅用于区分所描述的对象，不具有任何顺序或技术含义。而本申请所说“连接”、“联接”，如无特别说明，均包括直接和间接连接（联接）。

针对背景技术中提出的问题，现有技术中多采用对象存储系统（Object Storage System），对象存储系统作为一种具有高扩展性、高可靠性与高可用性、低成本的分布式存储系统，逐渐成为海量数据存储的良好选择。对象存储系统将数据实体作为具有唯一标识的单一对象（或将大数据实体切分为多个对象），通过节点选择算法将对象标识映射到集群中处于不同失效域的多个对象存储设备（Object Storage Device，以下简称OSD设备），对象以多副本或纠删码/再生码的方式冗余存储，从而实现高可靠性与高可用性。CEPH 作为一种对象存储系统（除此之外还支持文件级存储、块存储）设计了 CRUSH 算法，实现对象的均衡分布。CRUSH 采用一致性哈希（一种分布式哈希算法，Distributed Hashing Table，DHT）思想，首先将哈希空间（或称哈希环）划分为多个虚拟子空间（CEPH 中称为 Placement Group，PG），通过哈希函数，将对象标识映射到哈希环上某一子空间 PG 中，然后通过 CRUSH 算法，将 PG 映射到结构化集群中满足一定存储规则的 OSD 列表中。CRUSH 算法具有高可扩展性、伪随机均衡分布等优点，但也有其不足之处：（1）在建立集群节点架构时，只有上级节点到下级节点（序列）的包含关系，当涉及调节叶节点，即 OSD 设备的存储容量（权重）等参数时，从叶节点到根节点的整个遍历过程中，对于每一层级，都需要遍历整个集群找出子节点的父节点，时间复杂度为 **nlog(n)**，当集群规模较大时，效率较低；

（2）节点选择过程中，对选择点下属子节点的选取无限制，直接采用哈希值伪随机选择节点，一次选择过程结束后，再判断被选设备节点是否有效，如果候选节点与已选节点发生碰撞或失效，则需要重新选择，存在重复选择次数多，选择过程收敛较慢的问题；（3）从起始选择点到最底层设备叶节点的多层选择过程中，多次计算哈希，如当节点类型为 CRUSH_BUCKET_STRAW 时，每层的每个子节点都会计算哈希值，一定程度上降低了选择效率。

基于上述内容，在本发明申请实施例中，提出一种基于一致性哈希算法的分布式数据冗余存储方法，采用带监督的权重随机选择算法，从集群中为预存储数据选择目标存储节点过程中，只计算一次哈希用于将对象标识映射哈希子空间，提高节点选择效率，进而满足海量数据存储的要求。

**实施例一:**

本申请公开的一种基于一致性哈希算法的分布式数据冗余存储方法，包括:

步骤一、定义设备及节点属性，建立集群组织架构，以便于整个节点选择过程在结构化的集群组织图中实施，其执行流程具体包括:

1）.对设备及节点属性进行抽象，设备代表实际加入集群的存储设备。基于物理存储设备在集群组织架构中的位置定义节点。如图 1 所示，为基于物理存储设备构建的集群组织架构示意图，存储器（设备 Device 为一种特殊节点，位于集群架构最低层，为数据最终存储位置，与普通节点区别开来）一般具有唯一标识，其在集群组织架构中位置信息包括所属主机（Server）标识，机架（Rack）标识，数据中心标识等。其存储器相关属性，如存储容量（权重）等。其中权重代表此设备的相对存储容量，比如假设权重 1.0 代表 1T 容量，容量为 15T 的设备其权重值为 15.0。节点为集群组织架构的组成单元，以树形层级架构组织，其中根节点（Root）为顶层节点，其为虚拟实体，代表整个集群，常为节点选择起始点。节点（除设备层之外的中间层节点）可包括但不限于主机，机架，数据中心等，其属性包括但不限于节点标识（NodeId）、节点类型（NodeType）、节点权重（Weight）、子节点序列（ChildrenNodes）、子节点权重序列（ChildrenWeights）、父节点指针（ParentNode，设备节点也有父节点指针）等，其中节点权重等于下属所有子节点权重之和，即表示节点存储容量等于下层所有子节点存储容量之和；最底层叶节点，即存储器，包括所有存储器属性，其为节点选择算法的最终选择对象。假设在一个小型集群中，有设备 Device、主机 Server、机架 Rack 三个存储层级，增加虚拟节点 Root 在最高层，代表整个集群，因此 NodeType 的取值为:

{NODE_TYPE_DEVICE,

NODE_TYPE_SERVER,

NODE_TYPE_RACK,

NODE_TYPE_ROOT},

其中每 Server 可包含多于一个 Device，Device 的逻辑表现为挂载点或目录。为将实际存储节点区别开来，我们称存储层级最底层设备节点为设备（Device），其他存储层级为节点（Node）。

2）.将所有设备依次加入集群中，首先创建 Root 节点，代表整个集群；对每个设备，从其所属位置的最高层级开始，如数据中心或 Rack 等，判断节点（此待添加设备的祖先或父节点）是否存在，如不存在则创建此节点，设置其父节点（Root 的父节点为空），并设置其为下一个插入点，直到创建设备的直接父节点，并设置此父节点为将加入设备的插入点。

3）.创建设备节点，并添加到插入点，设置插入点为此设备的直接父节点。

4）.设置设备权重（或其它可能用于选择的属性）并修改其所有父节点（祖先节点）权重，满足父节点权重等于所有子节点权重之和，由前所述，本发明在节点属性中添加了直接父节点指针，从而使权重修改操作的时间复杂度由 **nlog(n)** 降低到 **log(n)**，加快了权重调整速度。以下具体举例说明。

如图 2 所示，为根据物理存储设备在集群组织架构中的位置定义节点的流程图，先创建 Root 节点，其为非实体节点，代表整个集群组织架构。则，

NodeId=0，NodeType=NODE_TYPE_ROOT, Weight=0,

子节点及其权重序列定义为

ChildrenNodes={}，ChildrenWeights={}

因此时无任何子节点，数组为空。

ParentNode=NULL（Root 节点位于存储层级最上层，无父节点）。

再将设备逐一加入集群，首先检查设备 DeviceId 是否已存在，若存在，则结束，继续添加下一设备。从设备所处位置的最高存储层级开始（本实施例中为 Rack），检查标识为（Device.RackId）的节点是否存在于集群中，若不存在，则创建新节点 RackNode，

NodeId=Device.RackId，

NodeType=NODE_TYPE_RACK,

Weight=0,

ChildrenNodes={},

ChildrenWeights={},

ParentNode=Root（父节点指针），

将 RackNode 添加进其父节点 Root 的子节点列表 Root.ChildrenNodes，将其权重 Weight 添加进 Root.ChildrenWeights，将 RackNode 设置为下一个新创建节点的插入点（即其 ParentNode）。

检查标识为 Device.ServerId 的节点是否存在于集群中，若不存在，则创建新节点 ServerNode，

NodeId=Device.ServerId，

NodeType=NODE_TYPE_SERVER，

Weight=0，

ChildrenNodes={}，

ChildrenWeights={}，

ParentNode=RackNode（父节点指针），

将 ServerNode 添加进其父节点 RackNode 的子节点列表中，

RackNode.ChildrenNodes，

将其权重 Weight 添加进 Rack.ChildrenWeights，

将 ServerNode 设置为下一个新创建节点的插入点。

当设备的所有上层节点创建完毕后，便得到设备在集群中的插入点，本实施例中为 ServerNode，创建新节点 DeviceNode，

NodeId=Device.DeviceId，

NodeType=NODE_TYPE_DEVICE，

Weight=0（后续步骤调整为 Device.Weight），

ChildrenWeights={}，

其父节点指针 ParentNode=ServerNode，

将 DeviceNode 添加进其父节点 ServerNode 的子节点列表中，

ServerNode.ChildrenNodes，

将其权重 Weight 添加进 Server.ChildrenWeights；

设备节点创建完毕后，需调整其上层节点的权重及子节点权重列表，调整上层子节点权重列表中子节点权重为对应下层子节点权重，本层节点权重等于其所有子节点权重和；本实施例中在节点结构中加入父节点指针，从而使权重的调整过程只需沿着父节点指针指向的路径，逐一修改父节点的权重 Weight 并将子节点 Weight 加入到其 ChildrenWeights 列表即可，时间复杂度为 $O(\log(n))$。

所有设备添加后，集群架构初始创建完毕，如图 1 所示，集群含有 2 个 Rack，每 Rack 有 2 个 Server，每 Server 有 2 个 Device。

步骤二、依据集群组织架构建立存储规则。依据预存储数据冗余存储方式，确定数据份数和每一份存储数据。

依据集群组织架构建立存储规则，作为后续权重随机选择算法的监督规则。先确定预存储数据的总副本数，然后确定每一存储层级允许存储的最大副本数，最后建立节点类型与最大存储副本数的映射关系表。其中，设定存储层级与节点类型相对应，映射关系表中包括为对象允许存储的最大副本数与对应的节点类型的节点允许被选择的最大次数相对应。其执行流程具体包括:

1）. 冗余存储方式采用多副本模式时，确定预存储数据的总副本数 N，此值与存储集群采用的数据冗余策略有关， N 为预存储数据的预设的总副本数，每一份存储数据为预存储数据或其副本。冗余存储方式采用纠删码(N, K)模式时，将预存储数据切分为 K 数据块，编码成 N 编码块， N 为对预存储数据编码后的总编码块数。数据份数则为总编码块数，每一份存储数据对应一个编码块。

2）.确定每一存储层级（即节点类型，包括主机、机架、数据中心、集群代表 Root 等）允许存储的最大副本数；这个由集群规模、失效域定义（如支持单/多主机、机架、数据中心失效等）、存储位置偏好（是否要求就近存放）等决定，部署时根据实际情况确定；对于一般情况，如采用多副本模式，Root 节点必然保存所有 N 副本，若其下层节点（即子节点）数为 M，则可设置其每个子节点最多存放⌈N/M⌉（不小于N/M 的整数）副本。

3）.建立存储规则，本实施例中定义一种存储规则为节点类型与最大存储副本数映射关系表，即 Key-Value Map，其中 Key 为节点类型， Value 为对象允许存储的最大副本数。失效域定义具体指支持单/多主机、机架和/或数据中心失效。数据冗余模式具体指多副本、纠删码和/或再生码。我们称某一规定长度的数据段为对象（Object），对象为数据副本或编码的基本单元，对象存储于哈希子空间中，后者为节点选择主体。建立存储规则的具体步骤，首先，确定预存储数据总副本数 N。在分布式存储系统中，通过增加数据冗余提高数据安全性。常用数据冗余方式有多副本、纠删码/再生码等，多副本方式保存对象的 N 份副本于集群不同位置，纠删码将对象切分为 K 数据块，编码后形成 N 编码块，然后将 N 编码块保存于集群不同位置。本示例中假设使用 3 副本模式，即 N=3。其次，确定每一存储层级（本实施例中，包括设备 Device，主机 Server、机架 Rack、Root）允许存储的最大副本数。每一层级最大副本数的确定，由集群规模、故障域划分及其数量、是否设置存储位置偏好等因素决定。本实施例考虑 3 副本模式的一般情况，规定一个 Rack 中保存的副本数不大于 2，每 Server 或 Device 最多保存一个副本。如下表所示，为节点选择监督规则表:

| 存储等级 | 类型名 | 类型值 | 允许存储最大副本数 | 备注 |
|---|---|---|---|---|
| Root | NODE_TYPE_ROOT | 3 | 3 | 副本总数 N |
| Rack | NODE_TYPE_RACK | 2 | 2 | |
| Server | NODE_TYPE_SERVER | 1 | 1 | |
| Device | NODE_TYPE_DEVICE | 0 | 1 | |

最后，建立节点类型与最大存储副本数映射关系表，设置:

RootTypeTimes=3,

RackTypeTimes=2,

ServerTypeTimes=DeviceTypeTimes=1。

因为 Root 代表整个集群，所以存放副本数目为 N=3。

步骤三、定义哈希环参数，依据所述存储规则，利用带监督的权重随机选择算法，从集群中为预存储数据选择目标存储节点序列。

在建立结构化集群组织图和存储限制规则后，便可执行带监督的一致性哈希节点权重随机选择算法。首先确定起始选择点，从起始选择点开始，对于每一存储层级，依据存储规则限制，根据节点权重值，随机选择合适节点，直到集群组织架构的叶节点，从叶节点中选择合适的设备节点。哈希环参数包括哈希空间长度、哈希子空间长度、哈希子空间数量和采用的哈希函数 Hash。哈希子空间长度对应同样长度的哈希值范围，邻接且不相交，共同组成整个哈希空间。预存储数据带有唯一的标识，通过计算其标识哈希值，映射到唯一哈希子空间。其执行流程包括:

1）.定义哈希环参数，包括哈希空间长度 L（即哈希值长度，如 32bits）、哈希子空间长度 SL、采用的哈希函数等，其中根据 L 和 SL 可得哈希子空间总数 $SN=2^{(L-SL)}$。

2）.根据对象标识 ObjectId 寻址其所属哈希子空间，其为对象标识的哈希值取模哈希子空间总数，即 Subspace=Hash(ObjectId)%SN。

3）.选择起始点，可从任意存储层级节点开始，对于初始选择，一般从 Root 节点开始，为 Subspace 选择 N 存储设备，因为 Subspace 为数据冗余基本单元，多个对象可映射进同一 Subspace。

4）.从选择点开始，对于每一存储层级，依据存储规则限制，根据子节点权重值，随机选择合适子节点，直到集群架构树的最底层叶节点，最后从叶节

点中选择合适的设备节点；本发明在此处的创新点在于，在每一层级选择时，首先判断子节点是否已达到存储规则限制，即被选择次数是否超过最大允许次数，若达到限制，则排除子节点，如果此层级所有子节点都已达到存储限制，则排除整个层级（即父节点）；当在最底层设备节点中选中合适的节点时，首先更新（加 1）其选中次数，然后更新其父路径上直到 Root 节点所有节点的被选择次数（子节点被选中代表其所有祖先节点也被选中）。简单理解为，起始选择点为选择算法的起始入口，从根节点开始，也可从任意存储层级节点开始，每一个存储层级指定所述节点被允许选择的最大次数。选择起始选择点从最高选择点开始，逐层往下选择，每一层被选出的节点作为新的入口点，重复层级节点选择过程，直到叶节点。若有合适设备被选中，则更新此设备及其所有父节点的被选择次数。判断子节点是否已达到存储规则限制，若达到限制，则排除此子节点；若此层级所有子节点都已达到存储限制，则排除子节点的父节点。在子节点列表中选择节点过程中，采用依据子节点的权重值进行选择。根据存储规则，先排除不符合条件的节点，不符合条件的节点具体可以指节点已达到存储限制。

5）.当已选择设备数等于指定值（如 N），或者尝试选择的次数大于某设置阈值时，则停止搜索，若无足够满足存储规则的设备（如集群规模设备数太少或集群存储分配不均衡），此时可从集群设备随机挑选仍欠缺数目设备节点，可以优先从未被选择的设备中选择。

以下举例说明，利用带监督的节点权重随机选择算法，从集群中为预存储数据选择目标存储节点（设备）序列。

首先，定义哈希环参数，本实施例中定义哈希空间长度 L=32bits，即哈希空间为：$[0, 2^{32}-1]$，每哈希子空间的长度为 SL=20bits，哈希子空间数量 $SN=2^{(32-20)}=2^{12}$，采用的哈希函数为 Murmur3，其输出哈希值为 32 位整数。然后计算对象哈希值，定位哈希子空间：

ObjectHash=Murmur3(ObjectId)，

利用哈希函数计算对象 ID 哈希值；

HashSubspaceValue=ObjectHash%SN，

定位哈希子空间，同一哈希子空间对应相同的目标设备列表。

如图 3 所示，为哈希环划分及对象路由过程示意图，展示哈希环划分及哈希子空间节点序列路由过程，其中每个对象对应一个哈希子空间，哈希子空间映射到多设备节点，每存储设备被分配多个哈希子空间。

然后，确定起始选择点， 可从集群任意点开始，在其下层级子节点序列中，

选择符合存储规则的指定数量的设备。本实施例中，假设从 Root 节点开始，为对象（即对应哈希子空间 HashSubspaceValue）选择 3 个设备节点。

最后，从选择点开始，对于每一存储层级，依据存储规则限制，根据节点权重值，随机选择合适节点，直到集群架构树的最底层叶节点，从叶节点中选择合适的节点（即存储设备），如图 4 所示，为节点选择总体流程图，具体实施步骤:

获取存储规则,即节点类型与节点最大被选择次数（本事实例中为存储最大副本数）映射关系表。

初始化节点标识 NodeId 与节点被选择次数映射表 SelectedNodeTimesMap，本示例从 Root 节点开始进行全新选择过程，则 SelectedNodeTimesMap 为空，即无任何已被选节点。

设置从选择点开始的要选择节点的数目 ReplicasNumber:

本实施例中，从 Root 开始，则 ReplicasNumber=N=3。

采用深度优先搜索方式，从 Root 节点开始，每选择一个下层节点，则以此节点为选择点，重复选择过程，直到选择一个合适的设备节点 Device，然后返回到起始选择点。根据存储规则带监督的选出一个节点，判断其节点类型是否为设备且节点标识有效。若是，则判断其被选择次数是否超过最大允许被选次数 TypeMaxSelectionNumber。若超过，则选择发生碰撞，返回无效节点标识 NODE_ITEM_INVALID。若未超过，更新 SelectedNodeTimesMap 中此节点标识的被选择次数，返回设备的节点标识 NodeId。若选择的节点类型为非设备，则以此节点为选择点，重复进入下一层级的节点选择过程。每一次节点选择过程返回时，若选择的节点标识不等于 NODE_ITEM_INVALID（无效节点），则更新 SelectedNodeTimesMap 中此节点的被选择次数;

进一步，每次成功选择一个设备节点时，都会更新此设备节点及其所有祖先节点直到 Root 节点的被选择次数，即更新监督规则;

当选择的设备节点数等于 ReplicasNumber，或者尝试选择的次数大于某设置阈值（如当集群规模较小时，设置为集群设备数目的 2 倍，实际可能只需略大于 ReplicasNumber 即可），则停止搜索，若无足够满足存储规则的设备（假设已选 ReplicasNumberSelected），此时可优先从集群未选设备中，随机挑选（ReplicasNumber-ReplicasNumberSelected）设备节点。

其中，每一层级的节点选择过程如图 5 所示，为带监督的权重随机节点选择流程图，其具体执行流程为:

获取选择点 SelectingNode 子节点 ChildrenNodes 的节点类型及存储规则中

限定的此子节点类型允许被选的最大次数 TypeMaxSelectionNumber，此即为本发明带监督节点选择算法的监督规则，每次选择范围限定在符合存储规则的候选子节点内，避免随机选择后再去判断节点是否为合适节点的过程，从而减少重复选择次数，加快选择收敛速度。

如果 SelectingNode 的子节点列表 ChildrenNodes 只有一个节点，则判断此子节点被选择次数是否超过此类型节点最大选择次数 TypeMaxSelectionNumber，若没有，则选择此唯一子节点，返回此子节点指针。否则，设置此 SelectingNode 的选择次数为 TypeMaxSelectionNumber，关闭此选择点。若 SelectingNode 的子节点列表 ChildrenNodes 存在多于一个节点，则继续。对于子节点列表 ChildrenNodes 中的每一个节点，计算其权重和 TotalWeight，此过程中排除已被选择次数超过其类型对应的最大选择次数 TypeMaxSelectionNumber 的节点。如果 TotalWeight=0，则无合适节点，返回空节点指针 NULL，否则，继续。生成随机数 RandomWeight，并对 TotalWeight 取模，即 RandomWeight=rand()%TotalWeight。依次累加 ChildrenNodes 中节点权重，直到权重和 HitSumWeight>=RandomWeight,记录此时被累加的子节点索引 ChildIndex。返回 ChildrenNodes[ChildIndex]，此为被选中节点。

基于上述实施例，本申请公开了一种基于一致性哈希算法的分布式数据冗余存储方法，本方法由于在集群节点中加入父节点指针，从而使节点属性设置或调整的效率提升，表现为时间复杂度由$n\log(n)$降低到$\log(n)$。同时在节点选择过程中，加入监督规则，事先排除不符合条件的节点，从而避免或减少随机选择后碰撞几率，减少重试选择次数，加快节点选择速度。另外每一层级子节点选择时，用面向每层级的（单次）随机函数取代面向每层级所有子节点的（多次）哈希值计算，进一步提高节点选择效率。本方法加快了存储设备属性调整时间，当集群节点状态发生变化时，如因故障导致离线、节点增加、节点移除时，可提供更快的响应速度，从而减少系统修复或负载均衡时间。另外由于节点选择效率的提升，加快了数据访问时对象寻址过程，从而可提高数据访问速度和存储系统吞吐量。

以上应用了具体个例对本发明进行阐述，只是用于帮助理解本发明，并不用以限制本发明。对于本发明所属技术领域的技术人员，依据本发明的思想，还可以做出若干简单推演、变形或替换。

1.一种基于一致性哈希算法的分布式数据冗余存储方法,其特征在于，该方法包括:

节点创建步骤：建立分布式存储系统中各存储节点的拓扑逻辑结构，并通过一致性哈希算法，确定各个哈希子空间所对应存储节点序列中各节点的位置;

数据写入步骤：依据预存储数据冗余存储方式，确定数据份数和每一份存储数据;

依据存储规则将所述每一份存储数据存储到不同的存储节点中。

2. 如权利要求 1 所述的方法，其特征在于，所述依据预存储数据冗余存储方式，确定数据份数和每一份存储数据包括:

所述冗余存储方式采用多副本模式时，所述数据份数为预设的副本数，所述每一份存储数据为预存储数据或所述预存储数据的副本;

所述冗余存储方式采用纠删码模式时，将所述预存储数据编码成编码块，所述数据份数为总编码块数，所述每一份存储数据对应一个编码块。

3.如权利要求 1 所述的方法，其特征在于，所述哈希环的参数包括哈希空间长度、哈希子空间长度、哈希子空间数量和采用的哈希函数 Hash;

所述哈希子空间长度对应同样长度的哈希值范围，邻接且不相交，共同组成整个哈希空间;

所述预存储数据带有唯一的标识，通过计算其标识哈希值，映射到唯一哈希子空间。

4. 如权利要求 1 所述的方法，其特征在于，所述建立分布式哈希环中哈希子空间与各存储节点的拓扑逻辑结构包括:

定义节点属性，其包括节点标识、节点类型、节点权重、子节点列表和子节点权重列表中至少一项；所述节点标识包括所述存储器的标识、所述主机的标识、所述机架的标识和所述数据中心标识中至少一项；所述节点类型包括存储器节点、主机节点、机架节点、数据中心节点和根节点中至少一项;

建立拓扑逻辑结构，基于物理存储设备集群组织架构构建拓扑逻辑结构，并根据所述物理存储设备在所述集群组织架构中的位置定义节点类型；所述物理存储设备包括存储器、控制所述存储器的主机、放置所述主机的机架和所述机架所属的数据中心;

定义所述根节点代表整个所述集群组织架构;

定义所述存储器为叶节点位于所述集群组织架构的最底层。

5.如权利要求 4 所述的方法，其特征在于，所述节点属性还包括父节点指

针；所述节点可沿所述父节点指针，逐级修改所述节点和所述节点的父节点的属性，直到根节点。

6. 如权利要求 4 所述的方法，其特征在于，所述依据存储规则将所述每一份预存储数据存储到不同的存储节点中包括：

确定每一存储层级允许存储的最大的数据份数；所述存储层级与所述节点的节点类型相对应；

建立所述节点的节点类型与所述最大的数据份数的映射关系表；所述映射关系表中包括所述允许存储的最大的数据份数与所述节点的节点类型允许被选择的最大次数相对应。

7. 如权利要求 6 所述的方法，其特征在于，还包括：

确定起始选择点；所述起始选择点为节点；

从所述起始选择点开始，依据所述存储规则限制和所述节点权重值，对所述起始选择点下每一存储层级依次选择子节点，直到选择到所述集群组织架构的叶节点，从所述叶节点中选择设备节点，用于存储所述每一份存储数据。

8. 如权利要求 7 所述的方法，其特征在于，

所述起始选择点从所述集群组织架构的最高层节点开始，逐层往下选择，每一层被选出的节点作为新的选择点，重复层级节点选择过程，直到选择到叶节点；在所述叶节点中选择设备节点；当选中设备节点时，更新所述设备节点及其所有被选择过的节点的父节点的被选择次数。

9. 如权利要求 8 所述的方法，其特征在于，还包括：

判断所述每一层被选出的节点的子节点是否已达到存储规则限制，若达到限制，则排除此子节点；若此层级所有子节点都已达到存储限制，则排除所述子节点的父节点；

在所述每一层被选出的节点的子节点列表中选择节点过程中，采用依据所述子节点的权重值进行选择。

10. 如权利要求 9 所述的方法，其特征在于，还包括：

根据所述存储规则，先排除不符合条件的节点；所述不符合条件的节点指所述节点已达到存储限制。

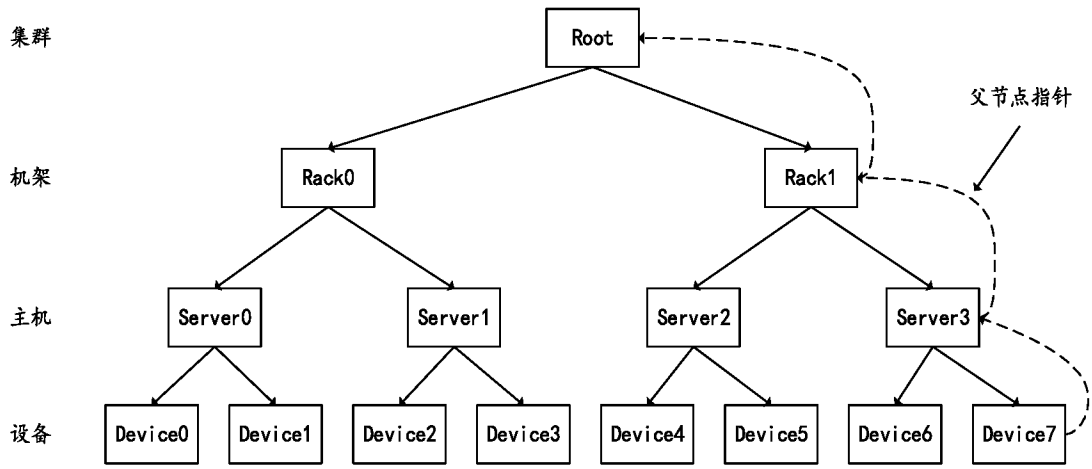

集群
机架
主机
设备
Root
父节点指针
Rack0
Rack1
Server0
Server1
Server2
Server3
Device0
Device1
Device2
Device3
Device4
Device5
Device6
Device7

图 1

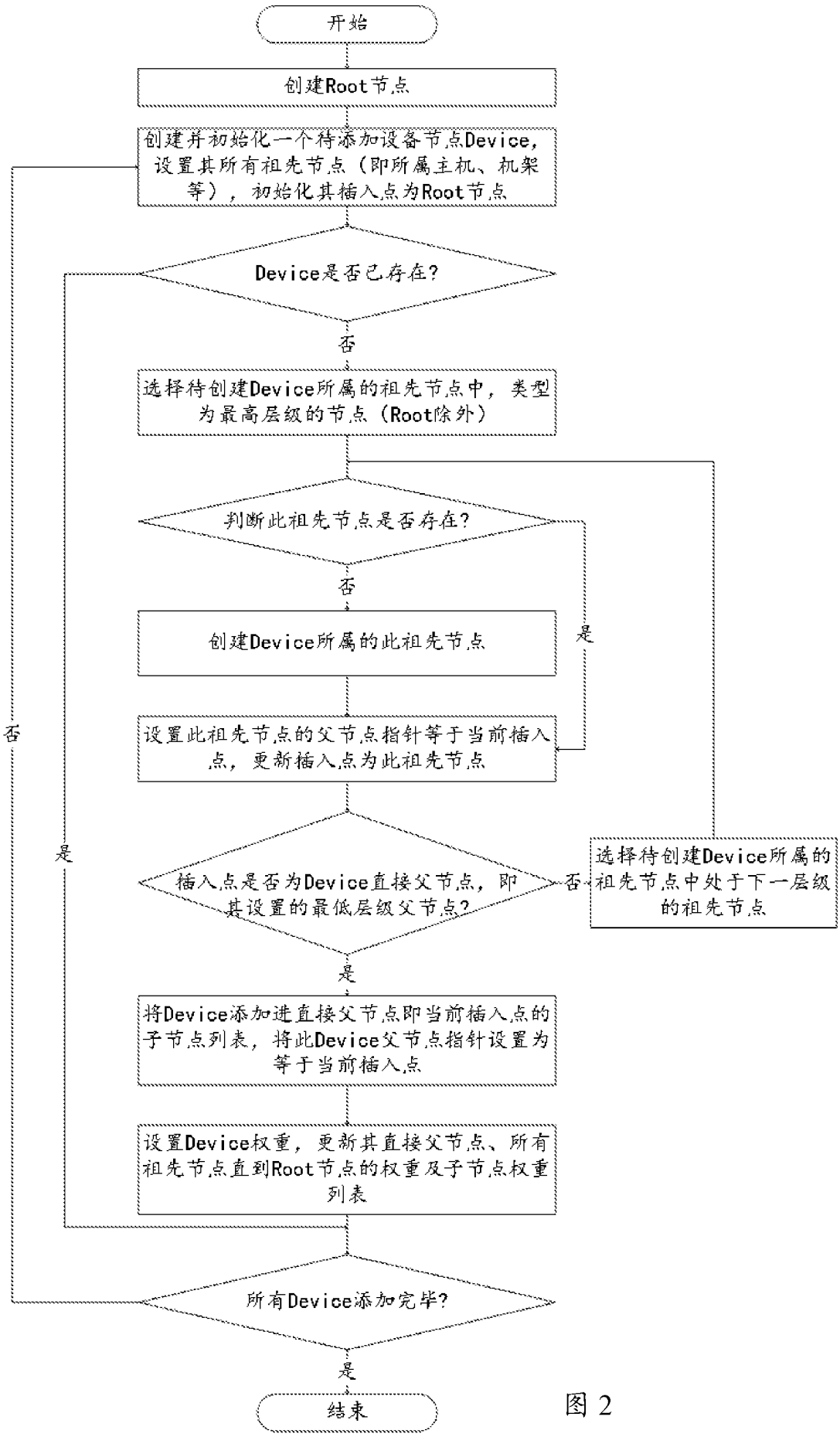

开始
创建Root节点
创建并初始化一个待添加设备节点Device，设置其所有祖先节点（即所属主机、机架等），初始化其插入点为Root节点
Device是否已存在?
否
选择待创建Device所属的祖先节点中，类型为最高层级的节点（Root除外）
判断此祖先节点是否存在?
否
创建Device所属的此祖先节点
是
设置此祖先节点的父节点指针等于当前插入点，更新插入点为此祖先节点
插入点是否为Device直接父节点，即其设置的最低层级父节点?
否
选择待创建Device所属的祖先节点中处于下一层级的祖先节点
是
将Device添加进直接父节点即当前插入点的子节点列表，将此Device父节点指针设置为等于当前插入点
设置Device权重，更新其直接父节点、所有祖先节点直到Root节点的权重及子节点权重列表
所有Device添加完毕?
否
是
是
结束

图 2

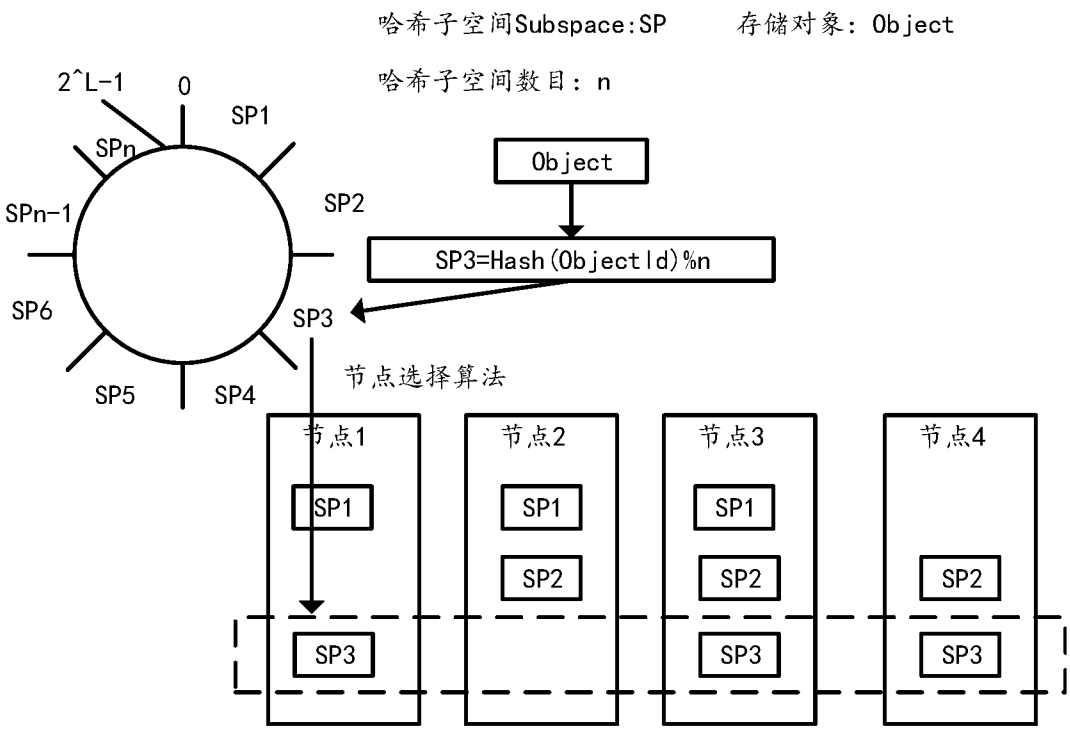

哈希子空间Subspace:SP
存储对象：Object
哈希子空间数目：n
2^L-1
0
SP1
SPn
SPn-1
SP2
SP6
SP3
SP5
SP4
Object
SP3=Hash(ObjectId)%n
节点选择算法
节点1
节点2
节点3
节点4
SP1
SP2
SP3

图 3

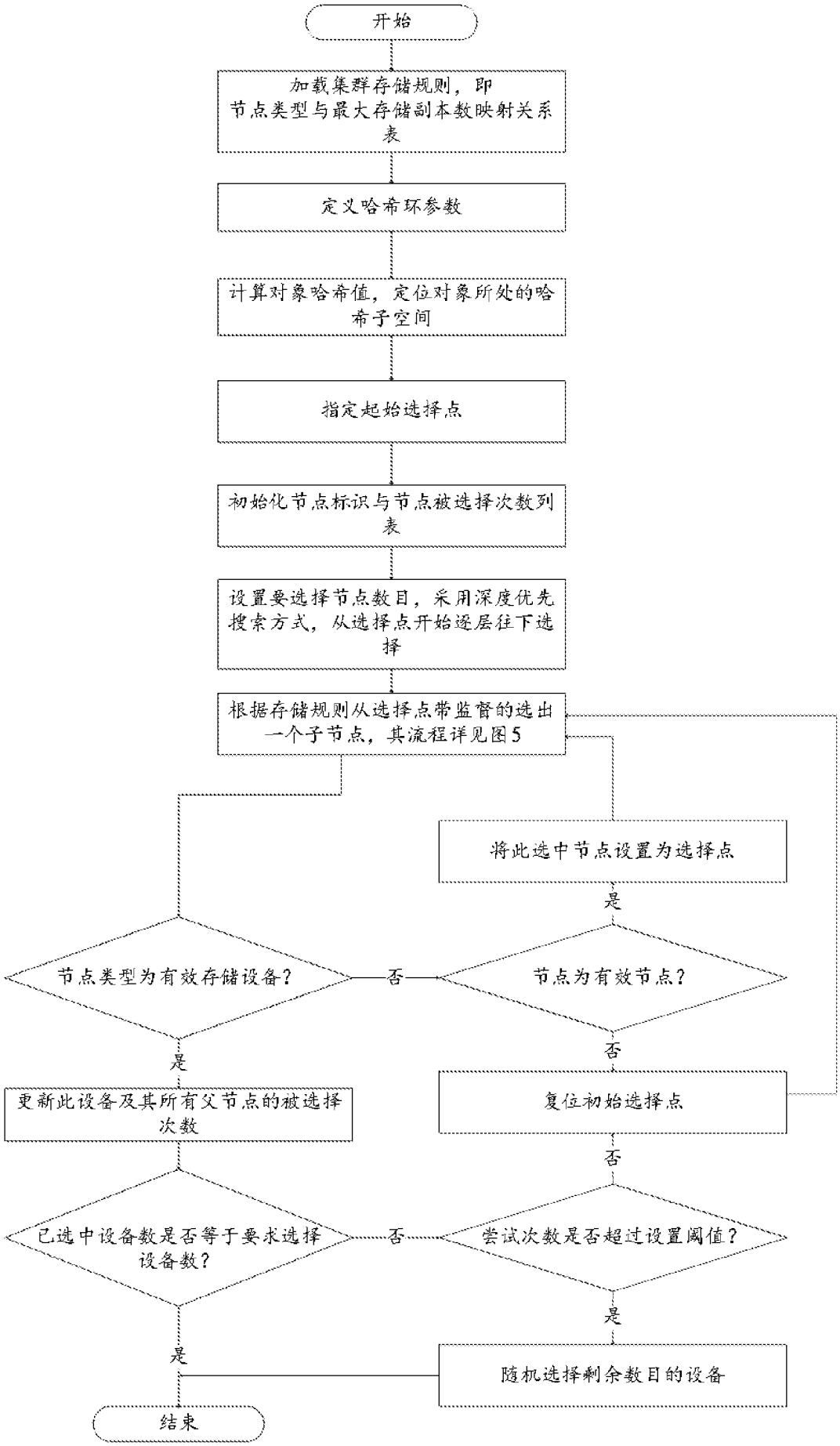

开始
加载集群存储规则，即节点类型与最大存储副本数映射关系表
定义哈希环参数
计算对象哈希值，定位对象所处的哈希子空间
指定起始选择点
初始化节点标识与节点被选择次数列表
设置要选择节点数目，采用深度优先搜索方式，从选择点开始逐层往下选择
根据存储规则从选择点带监督的选出一个子节点，其流程详见图5
将此选中节点设置为选择点
是
节点为有效节点？
否
节点类型为有效存储设备？
否
是
复位初始选择点
否
更新此设备及其所有父节点的被选择次数
尝试次数是否超过设置阈值？
否
已选中设备数是否等于要求选择设备数？
是
是
随机选择剩余数目的设备
结束

图 4

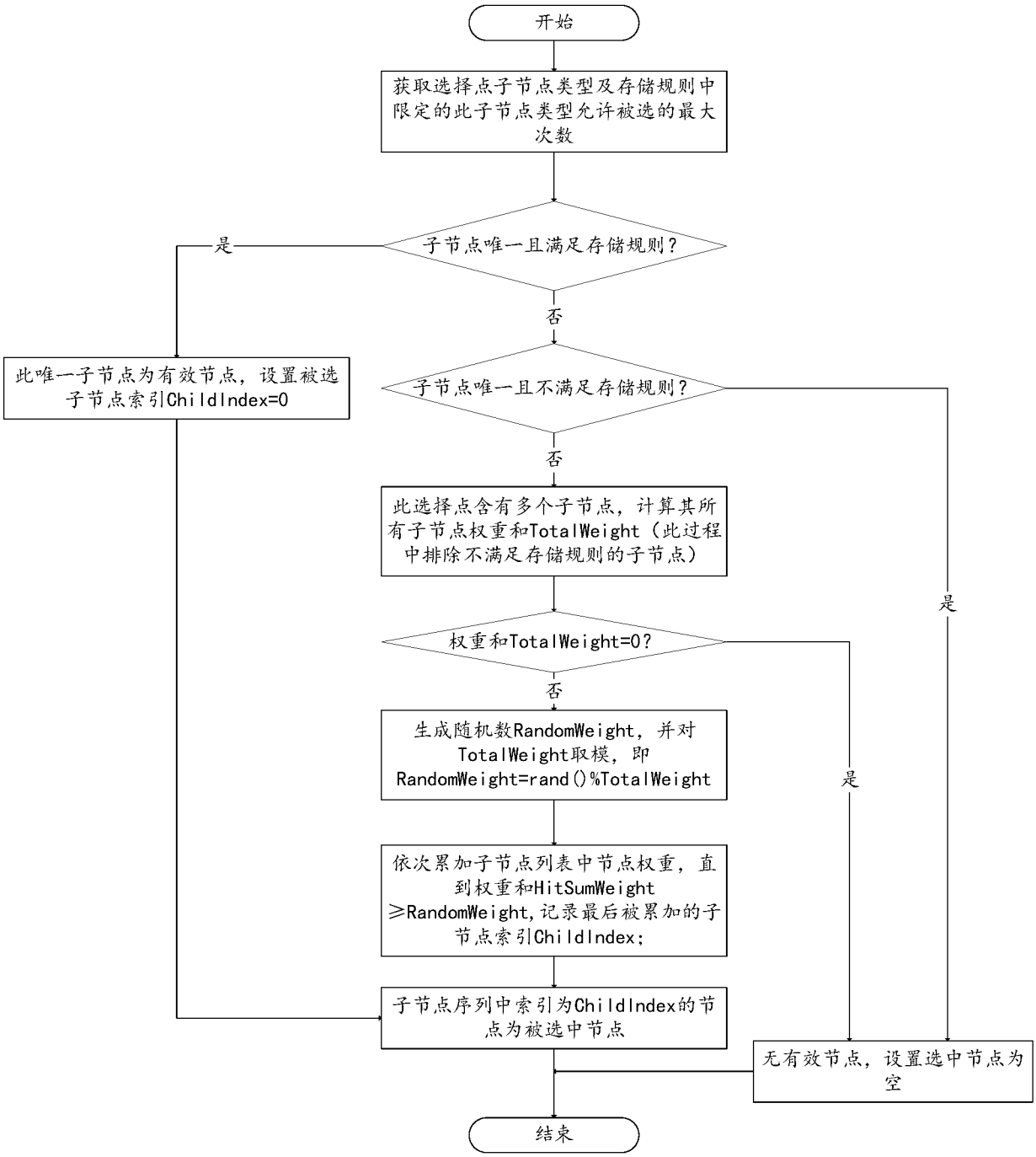

开始
获取选择点子节点类型及存储规则中限定的此子节点类型允许被选的最大次数
子节点唯一且满足存储规则？
是
否
此唯一子节点为有效节点，设置被选子节点索引ChildIndex=0
子节点唯一且不满足存储规则？
是
否
此选择点含有多个子节点，计算其所有子节点权重和TotalWeight（此过程中排除不满足存储规则的子节点）
权重和TotalWeight=0？
是
否
生成随机数RandomWeight，并对TotalWeight取模，即RandomWeight=rand()%TotalWeight
依次累加子节点列表中节点权重，直到权重和HitSumWeight≥RandomWeight,记录最后被累加的子节点索引ChildIndex;
子节点序列中索引为ChildIndex的节点为被选中节点
无有效节点，设置选中节点为空
结束

图 5

# INTERNATIONAL SEARCH REPORT

International application No.
**PCT/CN2018/095082**

## A. CLASSIFICATION OF SUBJECT MATTER

H04L 29/08(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L; G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT, WPI, EPODOC, CNKI, IEEE, GOOGLE: 一致性哈希, 分布式, 数据, 冗余, 存储, 节点, 子空间, consistent hash, distribute, data, redundant, store, node, subspace

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| A | CN 103561057 A (RESEARCH INSTITUTE OF TSINGHUA UNIVERSITY IN SHENZHEN) 05 February 2014 (2014-02-05)<br>description, paragraphs [0019]-[0024] | 1-10 |
| A | CN 102737130 A (GUANGZHOU CONGXING ELECTRONIC DEVELOPMENT CO., LTD.) 17 October 2012 (2012-10-17)<br>entire document | 1-10 |
| A | CN 103124299 A (HANGZHOU DIANZI UNIVERSITY) 29 May 2013 (2013-05-29)<br>entire document | 1-10 |
| A | US 2011231524 A1 (HITACHI, LTD.) 22 September 2011 (2011-09-22)<br>entire document | 1-10 |
| A | US 2017060865 A1 (INTERNATIONAL BUSINESS MACHINES CORPORATION) 02 March 2017 (2017-03-02)<br>entire document | 1-10 |

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| **20 March 2019** | **09 April 2019** |

| Name and mailing address of the ISA/CN | Authorized officer |
|---|---|
| **China National Intellectual Property Administration (ISA/CN)**<br>**No. 6, Xitucheng Road, Jimenqiao, Haidian District, Beijing 100088**<br>**China** | |
| Facsimile No. **(86-10)62019451** | Telephone No. |

**INTERNATIONAL SEARCH REPORT**
**Information on patent family members**

International application No.
**PCT/CN2018/095082**

| Patent document cited in search report | Publication date (day/month/year) | Patent family member(s) | Publication date (day/month/year) |
|---|---|---|---|
| CN 103561057 A | 05 February 2014 | None | |
| CN 102737130 A | 17 October 2012 | None | |
| CN 103124299 A | 29 May 2013 | None | |
| US 2011231524 A1 | 22 September 2011 | None | |
| US 2017060865 A1 | 02 March 2017 | None | |

# 国际检索报告

国际申请号
PCT/CN2018/095082

A. 主题的分类

H04L 29/08(2006.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

H04L; G06F

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用））

CNPAT,WPI,EPODOC,CNKI,IEEE,GOOGLE：一致性哈希，分布式，数据，冗余，存储，节点，子空间，consistent hash，distribute，data，redundant，store，node，subspace

C. 相关文件

| 类 型* | 引用文件，必要时，指明相关段落 | 相关的权利要求 |
|---|---|---|
| A | CN 103561057 A (深圳清华大学研究院) 2014年 2月 5日 (2014-02-05)<br>说明书第[0019]-[0024]段 | 1-10 |
| A | CN 102737130 A (广州从兴电子开发有限公司) 2012年 10月 17日 (2012-10-17)<br>全文 | 1-10 |
| A | CN 103124299 A (杭州电子科技大学) 2013年 5月 29日 (2013-05-29)<br>全文 | 1-10 |
| A | US 2011231524 A1 (HITACHI, LTD.) 2011年 9月 22日 (2011-09-22)<br>全文 | 1-10 |
| A | US 2017060865 A1 (INTERNATIONAL BUSINESS MACHINES CORPORATION) 2017年 3月 2日 (2017-03-02)<br>全文 | 1-10 |

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

“A” 认为不特别相关的表示了现有技术一般状态的文件

“E” 在国际申请日的当天或之后公布的在先申请或专利

“L” 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

“O” 涉及口头公开、使用、展览或其他方式公开的文件

“P” 公布日先于国际申请日但迟于所要求的优先权日的文件

“T” 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

“X” 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

“Y” 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

“&” 同族专利的文件

| 国际检索实际完成的日期 | 国际检索报告邮寄日期 |
|---|---|
| 2019年 3月 20日 | 2019年 4月 9日 |
| ISA/CN的名称和邮寄地址<br>中国国家知识产权局(ISA/CN)<br>中国北京市海淀区蓟门桥西土城路6号 100088<br>传真号 (86-10)62019451 | 受权官员<br>孔昕<br>电话号码 86-(10)-53961371 |

表 PCT/ISA/210 (第2页) (2015年1月)

# 国际检索报告
## 关于同族专利的信息

国际申请号
PCT/CN2018/095082

| 检索报告引用的专利文件 | 公布日<br>(年/月/日) | 同族专利 | 公布日<br>(年/月/日) |
|---|---|---|---|
| CN 103561057 A | 2014年 2月 5日 | 无 | |
| CN 102737130 A | 2012年 10月 17日 | 无 | |
| CN 103124299 A | 2013年 5月 29日 | 无 | |
| US 2011231524 A1 | 2011年 9月 22日 | 无 | |
| US 2017060865 A1 | 2017年 3月 2日 | 无 | |