(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 103259729 A
(43)申请公布日 2013.08.21

(21)申请号 201210525933.3

(22)申请日 2012.12.10

(71)申请人 上海德拓信息技术有限公司
地址 200233 上海市徐汇区桂箐路 7 号 3 号楼 203 室
申请人 浙江广播电视集团

(72)发明人 谢赟 吴新野 韩欣

(51)Int.Cl.
H04L 12/743(2013.01)
H04L 12/861(2013.01)

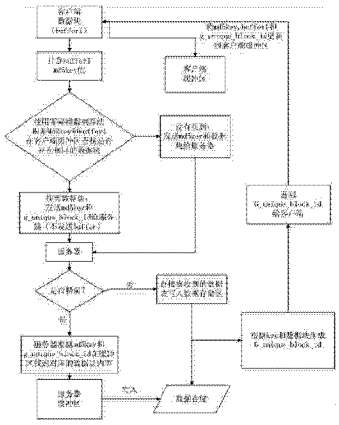
权利要求书2页 说明书4页 附图2页

(54)发明名称

基于零碰撞散列算法的网络数据精简传输方法

(57)摘要

本发明涉及用于局域网或广域网的数据传输领域，具体为一种基于零碰撞散列算法的网络数据精简传输方法。一种基于零碰撞散列算法的网络数据精简传输方法，包括将待传输的数据包分割，其特征是：还包括如下步骤：(1) 切分数据包；(2) 零碰撞散列操作；(3) 匹配数据块；(4) 写入存储区；(5) 更新缓冲区索引；(6) 写入存储区；(7) 重复第 (2) 至第 (6) 步。本发明数据精简率高，传输速率快，安全性强，适用范围广。



CN 103259729 A

1. 一种基于零碰撞散列算法的网络数据精简传输方法，包括将待传输的数据包分割，其特征是：还包括如下步骤：

(1) 在客户端对要传输的数据包实施切分，产生多组数据块，具体切分方法：设定一切分值，设切分值为 k_0，将数据包从第一个字节起算每 k_0 大小切分成一个数据块，如此对数据包依次切分直至剩下的数据块小于 k_0，将剩下的数据块作为最后一个数据块，完成切分；如果数据包小于 k_0，则不予切分而将整个数据包作为一个数据块；

(2) 对一个数据块实施零碰撞散列操作，设计算所得的散列值为 md5key，以 md5key 作为该数据块的指纹；

(3) 根据指纹和指纹所对应的数据块内容在客户端全局缓冲区查找是否存在相同的数据块；

(4) 如果客户端全局缓冲区不存在该数据块，则发送 md5key 和对应的数据块到服务器，服务器将 md5key 和数据块写入存储空间，并生成和 md5key 所对应的数据块标志，设数据块标志为 block_id，服务器更新服务器端全局缓冲区索引，同时将 block_id 返回给客户端；

(5) 客户端收到服务器返回的 block_id 后，将该数据块更新到客户端缓冲区索引，随后跳至第 (7) 步；

(6) 如果数据块在客户端索引区找到相同的数据块，则取得该数据块的 md5key 和 block_id，将 md5key 和 block_id 发送至服务器，服务器根据服务器端全局索引区找到 block_id 所对应的数据块，将该数据块写入存储空间；

(7) 对下一个数据块重复上述第 (2) 至第 (6) 步，直至数据包的所有数据块都被处理完毕，结束网络数据精简传输。

2. 如权利要求 1 所述的基于零碰撞散列算法的网络数据精简传输方法，其特征是：

第 (1) 步在客户端对要传输的数据包实施切分时，切分值为 k_0 为 512KByte；

第 (2) 步对一个数据块实施零碰撞散列操作时，零碰撞散列操作采用信息 - 摘要第 5 版算法。

3. 如权利要求 2 所述的基于零碰撞散列算法的网络数据精简传输方法，其特征是：

第 (2) 步对一个数据块实施零碰撞散列操作时，在客户端全局缓冲区存储该数据块的如下标志：md5key、block_count 和 block_id，其中，md5key 是该数据块的指纹，block_count 是和该数据块指纹相同的数据块的数量，block_id 是和该数据块指纹相同的各个数据块的逻辑块编号，存储时采用链表的形式，以 md5key 作为链表内每一条记录的标志，链表内的每条记录包括指纹信息、数量信息和逻辑块编号信息这三部分内容，链表的格式为：

指纹	数量	逻辑块编号
md5key(1)	block_count(1)	block_id(11)|block_id(12)|…|block_id(1x)
md5key(2)	block_count(2)	block_id(21)|block_id(22)|…|block_id(2y)
md5key(3)	block_count(3)	block_id(31)|block_id(32)|…|block_id(3z)
……	……	……
md5key(n)	block_count(n)	block_id(n1)|block_id(n2)|…|block_id(nm)

第 (3) 步根据指纹和指纹所对应的数据块内容在客户端全局缓冲区查找是否存在相同的数据块时，按如下步骤依次进行：

a. 计算待查数据块的散列值，设为 md5key，设待查数据块的全局逻辑块号为 block_id；

b. 根据 md5key 在链表中查找是否存在此散列值；

c. 如果不存在此散列值，则将和 md5key 所对应的 block_id 插入到链表的一条新记录中，并置 block_count 为 1；

d. 如果存在此散列值，设链表内已有 md5key(1)，且 md5key=md5key(1)，则根据 md5key(1) 所对应的所有的全局逻辑块号值，逐个寻址至各个数据块，将寻址所得数据块逐个地和待查数据块按位逐一比较，如果待查数据块和所有寻址所得数据块都不同，则将待查数据块的 block_id 补入 md5key(1) 记录的逻辑块编号信息部分，并将 md5key(1) 记录数量信息部分的 block_count(1) 加 1；

e. 如果待查数据块和寻址所得数据块中的一个相同，设待查数据块和数据块一相同，则取得数据块一的全局逻辑块号 block_id(11)，指定待查数据块的全局逻辑块号为 block_id(11)。

基于零碰撞散列算法的网络数据精简传输方法

技术领域

[0001] 本发明涉及用于局域网或广域网的数据传输领域，具体为一种基于零碰撞散列算法的网络数据精简传输方法。

背景技术

[0002] 一般在网络带宽限制的情况下为了节约带宽流量采用数据压缩和重复数据删除的数据缩减技术。采用数据压缩的方式一般使用 ZIP、RAR 等数据块压缩算法，在数据块从客户端通过网络传输到服务器端之前进行压缩，以缩短数据块长度，服务器端在收到数据后进行解压恢复原来的数据内容，从而实现节约带宽流量的目的。由于数据块内容重复的可能性很大，而压缩算法无法利用重复的数据块，因此精简效率不高。重复数据删除可以对存储空间数据进行有效的精简，通过删除其中重复的数据，只保留其中一份，从而消除冗余数据。重复数据删除技术多用于数据备份和归档场合，因为对数据进行多次备份后，存在大量的重复数据，非常适合使用这种技术。重复数据删除技术除了可以提高存储空间的利用率，也可以有效的减少网络数据传输。但是，重复数据块的识别技术大都采用数据块指纹(即 Finger Printer，简称 FP)技术，即通过散列算法计算数据块的散列值，将散列值作为数据块的指纹，常见的有 MD5、SHA-1、SHA-256、SHA-512 等，从纯数学角度看，如果两个数据块的指纹不同，则这两个数据块必然是不同的；然而，如果两个数据块指纹相同，仍不能断定这两个数据块是完全相同的，这是因为散列函数有可能会产生碰撞。不过，由于碰撞的概率非常小，并且可以通过提高散列位数的方法来进一步缩小碰撞概率，因此在近似条件下，可以认为数据块和指纹之间存在一一映射的关系。为了最大限度降低碰撞的概率，重复数据删除领域常用的 Bloom Filter 数据结构被设计成采用多种 hash 映射，希望能既降低冲突率又保证查询效率，但是无法从根本上解决问题，仍然存在一定的误识别率和删除索引困难的缺点，因此重复数据删除技术目前难以被用于关键数据存储场合。

发明内容

[0003] 为了克服现有技术的缺陷，提供一种数据精简率高、传输速率快、安全性强的网络数据传输方法，本发明公开了一种基于零碰撞散列算法的网络数据精简传输方法。

[0004] 本发明通过如下技术方案达到发明目的：

一种基于零碰撞散列算法的网络数据精简传输方法，包括将待传输的数据包分割，其特征是：还包括如下步骤：

(1) 在客户端对要传输的数据包实施切分，产生多组数据块，具体切分方法：设定一切分值，设切分值为 k_0，将数据包从第一个字节起算每 k_0 大小切分成一个数据块，如此对数据包依次切分直至剩下的数据块小于 k_0，将剩下的数据块作为最后一个数据块，完成切分；如果数据包小于 k_0，则不予切分而将整个数据包作为一个数据块；

(2) 对一个数据块实施零碰撞散列操作，设计算所得的散列值为 md5key，以 md5key 作为该数据块的指纹；

(3) 根据指纹和指纹所对应的数据块内容在客户端全局缓冲区查找是否存在相同的数据块；

(4) 如果客户端全局缓冲区不存在该数据块，则发送 md5key 和对应的数据块到服务器，服务器将 md5key 和数据块写入存储空间，并生成和 md5key 所对应的数据块标志，设数据块标志为 block_id，服务器更新服务器端全局缓冲区索引，同时将 block_id 返回给客户端；

(5) 客户端收到服务器返回的 block_id 后，将该数据块更新到客户端缓冲区索引，随后跳至第 (7) 步；

(6) 如果数据块在客户端索引区找到相同的数据块，则取得该数据块的 md5key 和 block_id，将 md5key 和 block_id 发送至服务器，服务器根据服务器端全局索引区找到 block_id 所对应的数据块，将该数据块写入存储空间；

(7) 对下一个数据块重复上述第 (2) 至第 (6) 步，直至数据包的所有数据块都被处理完毕，结束网络数据精简传输。

[0005] 所述的基于零碰撞散列算法的网络数据精简传输方法，其特征是：第 (2) 步对一个数据块实施零碰撞散列操作时，零碰撞散列操作采用信息 - 摘要第 5 版算法，即 Message-Digest Algorithm 5，简称 MD5。

[0006] 所述的基于零碰撞散列算法的网络数据精简传输方法，其特征是：

第 (1) 步在客户端对要传输的数据包实施切分时，切分值为 k_0 为 512KByte；

第 (2) 步对一个数据块实施零碰撞散列操作时，在客户端全局缓冲区存储该数据块的如下标志：md5key、block_count 和 block_id，其中，md5key 是该数据块的指纹，block_count 是和该数据块指纹相同的数据块的数量，block_id 是和该数据块指纹相同的各个数据块的逻辑块编号，block_id 由服务器全局索引区分配具有唯一性，存储时采用链表的形式，以 md5key 作为链表内每一条记录的标志，链表内的每条记录包括指纹信息、数量信息和逻辑块编号信息这三部分内容，链表的格式为：

指纹	数量	逻辑块编号
md5key(1)	block_count(1)	block_id(11)|block_id(12)|…|block_id(1x)
md5key(2)	block_count(2)	block_id(21)|block_id(22)|…|block_id(2y)
md5key(3)	block_count(3)	block_id(31)|block_id(32)|…|block_id(3z)
……	……	……
md5key(n)	block_count(n)	block_id(n1)|block_id(n2)|…|block_id(nm)

第 (3) 步根据指纹和指纹所对应的数据块内容在客户端全局缓冲区查找是否存在相同的数据块时，按如下步骤依次进行：

a. 计算待查数据块的散列值，设为 md5key，设待查数据块的全局逻辑块号为 block_id；

b. 根据 md5key 在链表中查找是否存在此散列值；

c. 如果不存在此散列值，则将和 md5key 所对应的 block_id 插入到链表的一条新记录中，并置 block_count 为 1；

d. 如果存在此散列值，设链表内已有 md5key(1)，且 md5key=md5key(1)，则根据 md5key(1) 所对应的所有的全局逻辑块号值，逐个寻址至各个数据块，将寻址所得数据块逐个地和待查数据块按位逐一比较，如果待查数据块和所有寻址所得数据块都不同，则将待查数据块的 block_id 补入 md5key(1) 记录的逻辑块编号信息部分，并将 md5key(1) 记录数

量信息部分的 block_count(1) 加 1 ;

e. 如果待查数据块和寻址所得数据块中的一个相同,设待查数据块和数据块一相同,则取得数据块一的全局逻辑块号 block_id(11),指定待查数据块的全局逻辑块号为 block_id(11)。

[0007] 本发明设计了一种使用零碰撞散列算法实现精简网络传输的方法,既保持了很好的传输精简率,又能确保不存在任何指纹冲突的情况,有效地解决了以往使用的压缩技术和重复数据删除技术的缺陷。可用于高并发的非结构化数据存储引擎中。

[0008] 本发明的有益效果是 :数据精简率高,传输速率快,安全性强,适用范围广。

附图说明

[0009] 图 1 是本发明的流程图 ;

图 2 是本发明在匹配数据块时的流程图。

具体实施方式

[0010] 以下通过具体实施例进一步说明本发明。

[0011] 实施例 1

一种基于零碰撞散列算法的网络数据精简传输方法,如图 1 所示,按如下步骤依次进行 :

(1) 在客户端对要传输的数据包实施切分,产生多组数据块,具体切分方法 :设定一切分值,设切分值为 k_0,将数据包从第一个字节起算每 k_0 大小切分成一个数据块,如此对数据包依次切分直至剩下的数据块小于 k_0,将剩下的数据块作为最后一个数据块,完成切分 ;如果数据包小于 k_0,则不予切分而将整个数据包作为一个数据块 ;本实施例中,k_0=512Kbyte ;

(2) 对一个数据块实施零碰撞散列操作,设计算所得的散列值为 md5key,以 md5key 作为该数据块的指纹 ;

(3) 根据指纹和指纹所对应的数据块内容在客户端全局缓冲区查找是否存在相同的数据块 ;

(4) 如果客户端全局缓冲区不存在该数据块,则发送 md5key 和对应的数据块到服务器,服务器将 md5key 和数据块写入存储空间,并生成和 md5key 所对应的数据块标志,设数据块标志为 block_id,服务器更新服务器端全局缓冲区索引,同时将 block_id 返回给客户端 ;

(5) 客户端收到服务器返回的 block_id 后,将该数据块更新到客户端缓冲区索引,随后跳至第 (7) 步 ;

(6) 如果数据块在客户端索引区找到相同的数据块,则取得该数据块的 md5key 和 block_id,将 md5key 和 block_id 发送至服务器,服务器根据服务器端全局索引区找到 block_id 所对应的数据块,将该数据块写入存储空间 ;

(7) 对下一个数据块重复上述第 (2) 至第 (6) 步,直至数据包的所有数据块都被处理完毕,结束网络数据精简传输。

[0012] 本实施例中,第 (2) 步对一个数据块实施零碰撞散列操作时,零碰撞散列操作采用信息 - 摘要第 5 版算法,即 Message-Digest Algorithm 5,简称 MD5,在客户端全局缓冲

区存储该数据块的如下标志：md5key、block_count 和 block_id，其中，md5key 是该数据块的指纹，block_count 是和该数据块指纹相同的数据块的数量，block_id 是和该数据块指纹相同的各个数据块的逻辑块编号，block_id 由服务器全局索引区分配具有唯一性，存储时采用链表的形式，以 md5key 作为链表内每一条记录的标志，链表内的每条记录包括指纹信息、数量信息和逻辑块编号信息这三部分内容，链表的格式为：

指纹	数量	逻辑块编号
md5key(1)	block_count(1)	block_id(11)|block_id(12)|…|block_id(1x)
md5key(2)	block_count(2)	block_id(21)|block_id(22)|…|block_id(2y)
md5key(3)	block_count(3)	block_id(31)|block_id(32)|…|block_id(3z)
……	……	……
md5key(n)	block_count(n)	block_id(n1)|block_id(n2)|…|block_id(nm)

本实施例中，第 (3) 步根据指纹和指纹所对应的数据块内容在客户端全局缓冲区查找是否存在相同的数据块时，如图 2 所示，按如下步骤依次进行：

a. 计算待查数据块的散列值，设为 md5key，设待查数据块的全局逻辑块号为 block_id；

b. 根据 md5key 在链表中查找是否存在此散列值；

c. 如果不存在此散列值，则将和 md5key 所对应的 block_id 插入到链表的一条新记录中，并置 block_count 为 1；

d. 如果存在此散列值，设链表内已有 md5key(1)，且 md5key=md5key(1)，则根据 md5key(1) 所对应的所有的全局逻辑块号值，逐个寻址至各个数据块，将寻址所得数据块逐个地和待查数据块按位逐一比较，如果待查数据块和所有寻址所得数据块都不同，则将待查数据块的 block_id 补入 md5key(1) 记录的逻辑块编号信息部分，并将 md5key(1) 记录数量信息部分的 block_count(1) 加 1；

e. 如果待查数据块和寻址所得数据块中的一个相同，设待查数据块和数据块一相同，则取得数据块一的全局逻辑块号 block_id(11)，指定待查数据块的全局逻辑块号为 block_id(11)。

[0013] 图 1 和图 2 中的 hashtable 指链表。

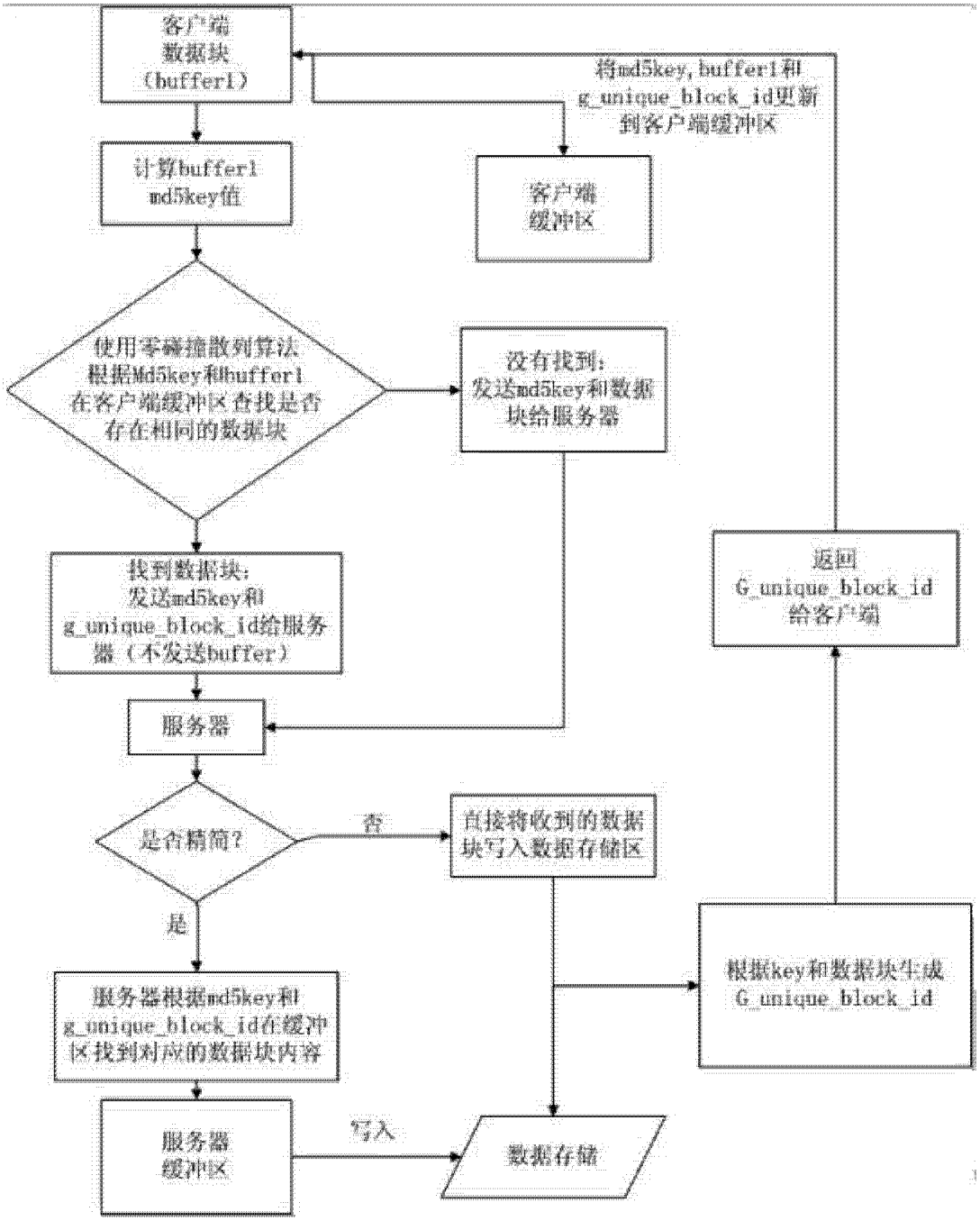

客户端
数据块
（buffer1）
计算buffer1
md5key值
使用零碰撞散列算法
根据Md5key和buffer1
在客户端缓冲区查找是否
存在相同的数据块
找到数据块：
发送md5key和
g_unique_block_id给服务
器（不发送buffer）
服务器
是否精简？
否
是
服务器根据md5key和
g_unique_block_id在缓冲
区找到对应的数据块内容
服务器
缓冲区
写入
数据存储
客户端
缓冲区
将md5key,buffer1和
g_unique_block_id更新
到客户端缓冲区
没有找到：
发送md5key和数据
块给服务器
直接将收到的数据
块写入数据存储区
根据key和数据块生成
G_unique_block_id
返回
G_unique_block_id
给客户端

图 1

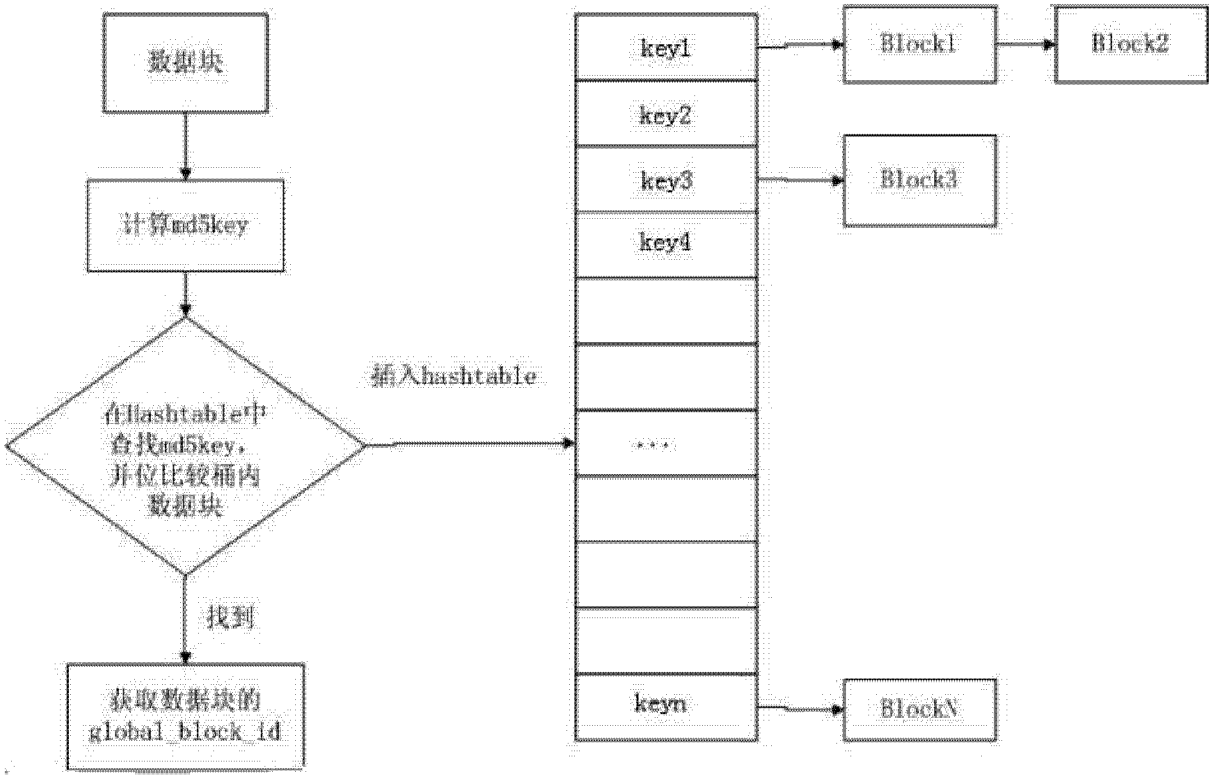

数据块
计算md5key
在Hashtable中
查找md5key，
并位比较桶内
数据块
找到
获取数据块的
global_block_id
插入hashtable
key1
key2
key3
key4
keyn
Block1
Block2
Block3
BlockN

图 2