



(12) 发明专利申请

(10) 申请公布号 CN 103902577 A

(43) 申请公布日 2014. 07. 02

(21) 申请号 201210579135. 9

(22) 申请日 2012. 12. 27

(71) 申请人 中国移动通信集团四川有限公司
地址 610041 四川省成都市高升桥路 1 号

(72) 发明人 曾健 陈刚 梅松 赵勇 马军

(74) 专利代理机构 北京中誉威圣知识产权代理有限公司 11279

代理人 郭振兴 丛芳

(51) Int. Cl.

G06F 17/30 (2006. 01)

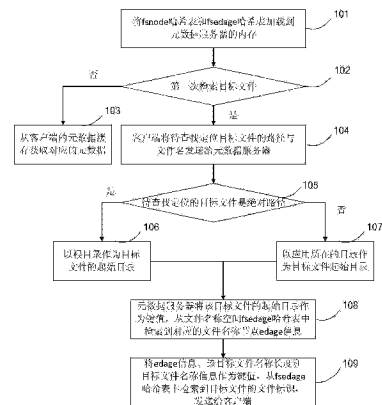
权利要求书2页 说明书4页 附图2页

(54) 发明名称

一种资源查找定位的方法和系统

(57) 摘要

本发明公开了一种资源查找定位的方法和系统,适用于云计算平台,该方法包括元数据服务器将文件节点信息的哈希值存储在 fsnode 哈希表中,将文件名称空间信息的哈希值存储在 fsedage 哈希表中;客户端将待查找定位的目标文件的路径与文件名发送给元数据服务器;元数据服务器将目标文件的起始目录作为键值,从 fsedage 哈希表中检索到对应的 edage 信息;将 edage 信息、目标文件名称长度和目标文件名称信息作为键值,从 fsedage 哈希表中检索到对应的目标文件的文件标识,发送给客户端。本发明的技术方案由于客户端同元数据服务器之间一次交互全部路径信息,可以减少检索次数和交互次数,同时通过客户端缓存元数据信息,减少客户端与元数据服务器之间的交互次数。



1. 一种资源查找定位的方法,适用于云计算平台,其特征在于,包括以下步骤:

元数据服务器将文件节点信息的哈希值存储在文件节点空间 fsnode 哈希表中,将文件名称空间信息的哈希值存储在文件名称空间 fsedage 哈希表中;

客户端将待查找定位的目标文件的路径与文件名发送给元数据服务器;

元数据服务器将所述目标文件的起始目录作为键值,从文件名称空间 fsedage 哈希表中检索到对应的文件名称节点 edage 信息;

元数据服务器将所述文件名称节点 edage 信息、所述目标文件名称长度和目标文件名称信息作为键值,从文件名称空间 fsedage 哈希表中检索到对应的目标文件的文件标识,发送给客户端。

2. 根据权利要求 1 所述的一种资源查找定位的方法,其特征在于,所述文件节点空间 fsnode 哈希表是文件节点 node 组成的哈希表,其中文件节点 node 包括文件类型、文件大小、数据存储位置和文件名称。

3. 根据权利要求 1 所述的一种资源查找定位的方法,其特征在于,文件名称空间 fsedage 哈希表是文件名称节点 edage 组成的哈希表,其中文件名称节点 edage 包含文件名称和文件标识。

4. 根据权利要求 1 所述的一种资源查找定位的方法,其特征在于,还包括以下步骤:

元数据服务器将文件节点空间 fsnode 哈希表和文件名称空间 fsedage 哈希表加载到元数据服务器的内存。

5. 根据权利要求 1 所述的一种资源查找定位的方法,其特征在于,还包括以下步骤:

客户端收到元数据服务器发送的元数据,存储在客户端的元数据缓存中。

6. 根据权利要求 5 所述的一种资源查找定位的方法,其特征在于,所述元数据包括文件节点信息和名称信息以及文件名称空间信息。

7. 根据权利要求 5 所述的一种资源查找定位的方法,其特征在于,所述元数据缓存格式包括文件绝对路径名、写引用数、读引用数、文件缓存修改标记、原文件大小、文件现在的大小、文件原分条数、文件现在的分条数、文件所有的分条信息和指向前后结构体的指针。

8. 根据权利要求 5 所述的一种资源查找定位的方法,其特征在于,所述客户端将待查找定位的目标文件的路径与文件名发送给元数据服务器,进一步包括以下步骤:

客户端判断待查找定位的目标文件是否第一次检索,如果是第一次,客户端将待查找定位的目标文件的路径与文件名发送给元数据服务器,如果不是第一次,从客户端的元数据缓存中获取对应的元数据。

9. 根据权利要求 1-8 中任一权利要求所述的一种资源查找定位的方法,其特征在于,还包括以下步骤:

元数据服务器判断所述待查找定位的目标文件使用的是绝对路径还是相对路径,如果是绝对路径,则以根目录作为所述目标文件的起始目录,如果是相对路径,则以应用所在的目录作为所述目标文件的起始目录。

10. 一种资源查找定位的系统,其特征在于,包括元数据服务器和客户端,其中元数据服务器用于将文件节点信息的哈希值存储在文件节点空间 fsnode 哈希表中,将文件名称空间信息的哈希值存储在文件名称空间 fsedage 哈希表中,将待查找定位的目标文件的起始目录作为键值,从文件名称空间 fsedage 哈希表中检索到对应的文件名称节点 edage 信

息,将所述文件名称节点 edage 信息、所述目标文件名称长度和目标文件名称信息作为键值,从文件名称空间 fsedage 哈希表中检索到对应的目标文件的文件标识,发送给客户端;

客户端用于将待查找定位的目标文件的路径与文件名发送给元数据服务器,并接收元数据服务器发送的目标文件的文件标识。

11. 根据权利要求 10 所述一种资源查找定位的系统,其特征在于,所述元数据服务器包括内存,所述内存用于存储文件节点空间 fsnode 哈希表和文件名称空间 fsedage 哈希表。

12. 根据权利要求 10 所述一种资源查找定位的系统,其特征在于,所述客户端包括缓存,所述缓存用于存储元数据。

一种资源查找定位的方法和系统

技术领域

[0001] 本发明涉及云计算技术领域,尤其涉及一种资源查找定位的方法和系统。

背景技术

[0002] 云计算平台上,现有文件资源查找定位方案是首先判断要检索的目标文件使用的是绝对路径还是相对路径。如果是绝对路径,那么将搜索的起始路径设置为根目录,从根目录开始检索。如果要检索的目标文件使用的是相对路径,则将搜索的起始路径设置为当前目录。

[0003] 现有方案会使用目录分隔符 (“\”或“/”)把要定位的目标按照目录的层次剥离开,分层地定位每一层的目标目录。当定位到第一层目录目标后,把第一层目录作为下次开始检索的起始目录,用来定位第二层目标目录,如此循环下去,直到定位到最终目标。

[0004] 例如 :/aaa/bbb/ccc/ddd 的定位是 :

[0005] 1、首先判断出使用的是绝对路径,那么把根目录作为检索的起始目录。

[0006] 2、使用分隔符 “/”,依照层次剥离出第一层定位目标 :aaa ;第二层定位目标 :bbb,第三层定位目标 :ccc ;第四层定位目标 :ddd。

[0007] 3、在根目录下首先定检索 aaa 目标。如果定位到 aaa,则把 aaa 作为下次检索的起始目录 ;如果没有定位到 aaa,那么整个检索操作结束,定位失败。

[0008] 4、在 aaa 目录下检索 bbb 目标。如果定位到 bbb,则把 bbb 作为下次检索的起始目录 ;如果没有定位到 bbb,那么整个检索操作结束,定位失败。

[0009] 5、在 bbb 目录下检索 ccc 目标。如果定位到 ccc,则把 ccc 作为下次检索的起始目录 ;如果没有定位到 ccc,那么整个检索操作结束,定位失败。

[0010] 6、在 ccc 目录下检索 ddd 目标。如果定位到 bbb,整个检索操作结束,定位成功 ;如果没有定位到 bbb,那么整个检索操作结束,定位失败。

[0011] 该技术方案存在的问题是网络交互次数多和检索次数多,分布式文件系统客户端使用目录间隔符划分开每一层检索目录,检索目标有多层目录,客户端就要与元数据服务器端有多次数据交互。

发明内容

[0012] 为了解决现有技术中存在的客户端与元数据服务器端之间数据交互次数过多的技术问题,本发明提出一种资源查找定位的方法和系统,能够减少检索次数和交互次数,减少客户端同元数据服务器端交互的次数。

[0013] 本发明一方面提供了一种资源查找定位的方法,适用于云计算平台,包括以下步骤 :

[0014] 元数据服务器将文件节点信息的哈希值存储在文件节点空间 fsnode 哈希表中,将文件名称空间信息的哈希值存储在文件名称空间 fsedage 哈希表中 ;

[0015] 客户端将待查找定位的目标文件的路径与文件名发送给元数据服务器 ;

[0016] 元数据服务器将所述目标文件的起始目录作为键值,从文件名称空间 fsedage 哈希表中检索到对应的文件名称节点 edage 信息;

[0017] 元数据服务器将所述文件名称节点 edage 信息、所述目标文件名称长度和目标文件名称信息作为键值,从文件名称空间 fsedage 哈希表中检索到对应的目标文件的文件标识,发送给客户端。

[0018] 本发明的另一个方面提供一种资源查找定位的系统,包括元数据服务器和客户端,其中元数据服务器用于将文件节点信息的哈希值存储在文件节点空间 fsnode 哈希表中,将文件名称空间信息的哈希值存储在文件名称空间 fsedage 哈希表中,将待查找定位的目标文件的起始目录作为键值,从文件名称空间 fsedage 哈希表中检索到对应的文件名称节点 edage 信息,将所述文件名称节点 edage 信息、所述目标文件名称长度和目标文件名称信息作为键值,从文件名称空间 fsedage 哈希表中检索到对应的目标文件的文件标识,发送给客户端;

[0019] 客户端用于将待查找定位的目标文件的路径与文件名发送给元数据服务器,并接收元数据服务器发送的目标文件的文件标识。

[0020] 本发明的技术方案由于客户端同元数据服务器之间一次交互全部路径信息,可以减少检索次数和交互次数,同时通过客户端缓存元数据信息,减少客户端与元数据服务器之间的交互次数。

附图说明

[0021] 图 1 是本发明实施例一中资源查找定位的流程图。

[0022] 图 2 是本发明实施例二中资源查找定位系统的结构示意图。

具体实施方式

[0023] 下面结合附图对本发明的具体实施方式进行详细描述。

[0024] 分布式文件系统由元数据服务器管理文件系统的元数据(包括文件目录树组织、属性维护、文件操作日志记录、授权访问等),管理整个存储系统的命名空间,对外提供单一的系统映像,并负责整个存储集群的管理监控。用来管理整个文件系统,存储每个文件的元数据(文件大小,文件属性,文件所在位置的这些信息),同时也包含了所有非规则文件的全部信息,如文件夹,套接字设备,管道设备。元数据服务器在初始化时,首先将所有的文件系统元数据信息加载到内存,主要包括:

[0025] Fsnode:文件节点空间,该数据结构是文件节点 node 组成的一个哈希表。

[0026] Fsedage:文件名称空间,该数据结构是文件名称 edage 组成的一个哈希表,其中 edage 包含元素有,文件名称,文件标识。

[0027] chunk:存储块相关信息包含 fsnode(节点)。

[0028] 文件系统中每一个文件作为系统的一个节点,所有节点组成一个链表,构成文件系统的节点空间。

[0029] 文件系统中每一个文件拥有文件名称,所有的文件名称组成一个链表,构成文件系统的名称空间所有的文件系统节点采用 hash 算法进行定位查找。

[0030] 图 1 是本发明实施例一中资源查找定位的流程图。如图 1 所示,该适用于云计算

平台的资源查找定位的流程包括以下步骤：

[0031] 步骤 101、元数据服务器启动时,将文件节点空间 fsnode 哈希表和文件名称空间 fsedage 哈希表加载到元数据服务器的内存。

[0032] 该文件节点空间 fsnode 哈希表是文件节点 node 组成的哈希表,其中文件节点 node 包括文件类型、文件大小、数据存储位置和文件名称。

[0033] 该文件名称空间 fsedage 哈希表是文件名称节点 edage 组成的哈希表,其中文件名称节点 edage 包含文件名称和文件标识。

[0034] 元数据服务器将所有的文件系统的元数据信息加载到内存,所有文件节点,以文件的标识为 key,将文件节点信息的哈希值存储在文件节点空间 fsnode 哈希表中,所有文件名称空间以其父节点标识、名称长度、名称信息为 key,将文件名称空间信息的哈希值存储在文件名称空间 fsedage 哈希表中。

[0035] 步骤 102、分布式文件系统的客户端访问分布式文件系统中的文件时,客户端判断待查找定位的目标文件是否第一次检索,如果是第一次,转至步骤 104、如果不是第一次,则转至步骤 103。

[0036] 步骤 103、从客户端的元数据缓存中获取对应的元数据,流程结束。

[0037] 元数据缓存队列是一个双向队列,队列上的每个节点保存的是某个文件的元数据缓存,元数据缓存格式包括文件绝对路径名、写引用数、读引用数、文件缓存修改标记、原文件大小、文件现在的大小、文件原分条数、文件现在的分条数、文件所有的分条信息和指向前后结构体的指针,具体的元数据缓存格式如下：

[0038] typedef struct meta_cache {

[0039] FILENAME path ;// 文件绝对路径名

[0040] int wcount ;// 写引用数

[0041] int rcount ;// 读引用数

[0042] char isdirty ;//0 表示该文件缓存没被修改过,否则表示修改过

[0043] uint64_t oldsize ;// 原文件 (从元数据服务器获取时) 大小

[0044] uint64_t newsize ;// 文件现在的大小

[0045] uint32_t o_chunknum ;// 文件原分条数

[0046] uint32_t n_chunknum ;// 文件现在的分条数

[0047] CHUNK_MEM**chunkarray ;// 文件所有的分条信息

[0048] struct meta_cache*next, *pre ;// 指向前后结构体的指针

[0049] }META_CACHE ;

[0050] 否则,客户端先从元数据服务器获取到该文件的元数据,该元数据包括文件节点信息和名称信息以及文件名称空间信息,初始化之后便插入客户端元数据缓存中,以后再打开该文件,则只需从本地元数据缓存中获取即可。

[0051] 步骤 104、客户端通过网络将待查找定位的目标文件的路径与文件名发送给分布式文件系统的元数据服务器。

[0052] 步骤 105、元数据服务器判断待查找定位的目标文件使用的是绝对路径还是相对路径,如果是绝对路径,则转至步骤 106 ;如果是相对路径,则转至步骤 107。

[0053] 步骤 106、以根目录作为目标文件的起始目录,并转至步骤 108。

[0054] 步骤 107、以应用所在的目录作为目标文件的起始目录,并转至步骤 108。

[0055] 步骤 108、元数据服务器将该目标文件的起始目录作为键值,从文件名称空间 fsedage 哈希表中检索到对应的文件名称节点 edage 信息。

[0056] 步骤 109、元数据服务器将该文件名称节点 edage 信息、该目标文件名称长度和目标文件名称信息作为键值,从文件名称空间 fsedage 哈希表中检索到对应的目标文件的文件标识,发送给客户端。

[0057] 例如 :/aaa/bbb/ccc/ddd 的查找定位 :

[0058] 1、从客户端元数据缓存读取 /aaa/bbb/ccc/ddd 文件的元数据,如果客户端元数据缓存中存在,从客户端元数据缓存直接存取 /aaa/bbb/ccc/ddd 文件的元数据 ;

[0059] 2、如果缓存中没有,则判断出使用的是绝对路径还是相对路径,把全路径 /aaa/bbb/ccc/ddd 作为检索信息,一次进行查找定位。

[0060] 为了实现上述流程,本发明实施例还提供了一种资源查找定位系统。图 2 是本发明实施例二中资源查找定位系统的结构示意图。如图 2 所示,该系统包括元数据服务器 21 和客户端 22,其中元数据服务器包括内存 211,客户端包括缓存 221。

[0061] 元数据服务器用于将文件节点信息的哈希值存储在文件节点空间 fsnode 哈希表中,将文件名称空间信息的哈希值存储在文件名称空间 fsedage 哈希表中,将待查找定位的目标文件的起始目录作为键值,从文件名称空间 fsedage 哈希表中检索到对应的文件名称节点 edage 信息,将所述文件名称节点 edage 信息、所述目标文件名称长度和目标文件名称信息作为键值,从文件名称空间 fsedage 哈希表中检索到对应的目标文件的文件标识,发送给客户端。其中内存用于存储文件节点空间 fsnode 哈希表和文件名称空间 fsedage 哈希表。

[0062] 客户端用于将待查找定位的目标文件的路径与文件名发送给元数据服务器,并接收元数据服务器发送的目标文件的文件标识。其中缓存用于存储元数据。

[0063] 本发明实施例中的技术方案由于客户端同元数据服务器之间一次交互全部路径信息,可以减少检索次数和交互次数,同时通过客户端缓存元数据信息,减少客户端与元数据服务器之间的交互次数。

[0064] 应说明的是 :以上实施例仅用以说明本发明而非限制,本发明也并不仅限于上述举例,一切不脱离本发明的精神和范围的技术方案及其改进,其均应涵盖在本发明的权利要求范围中。

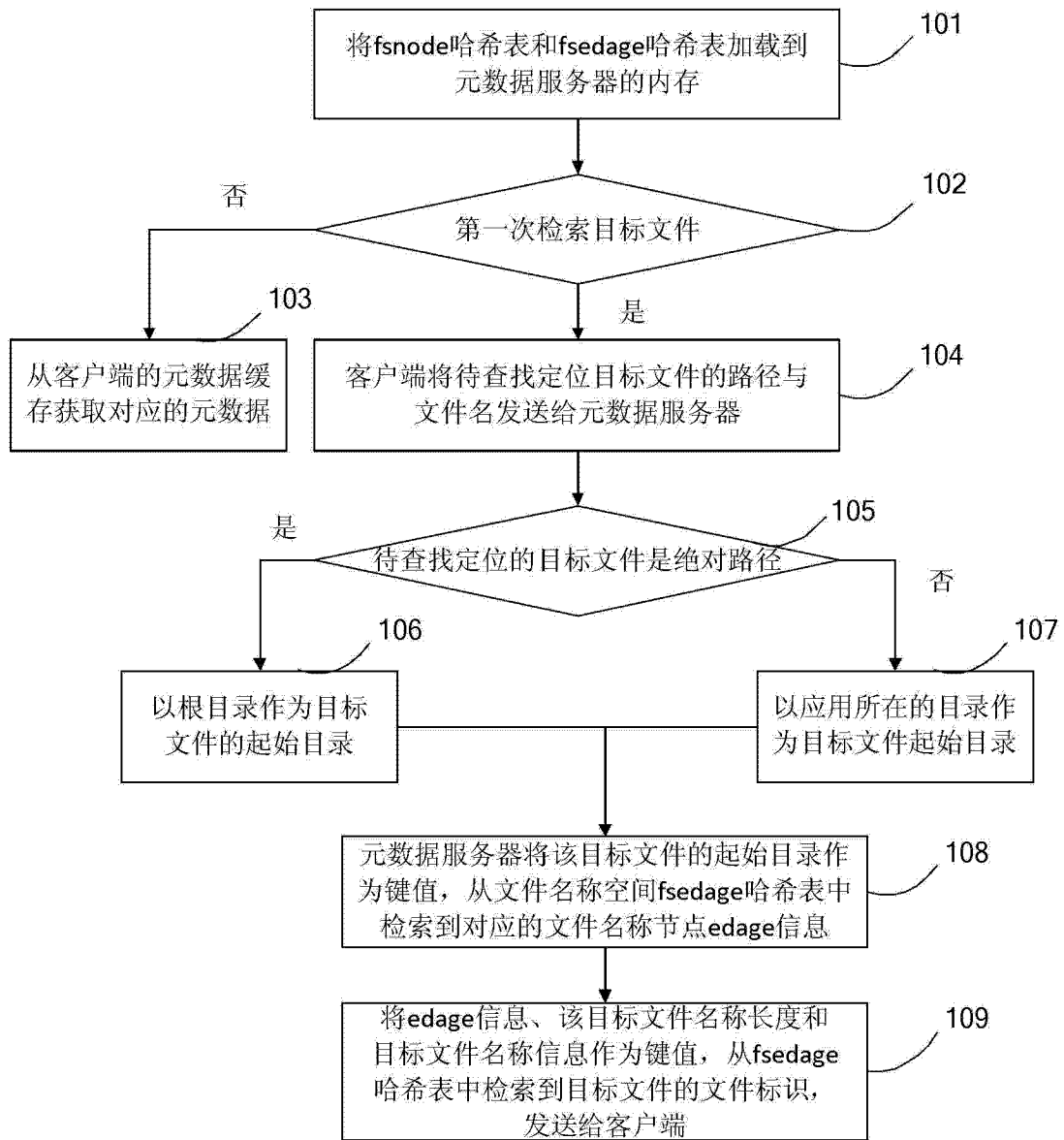


图 1

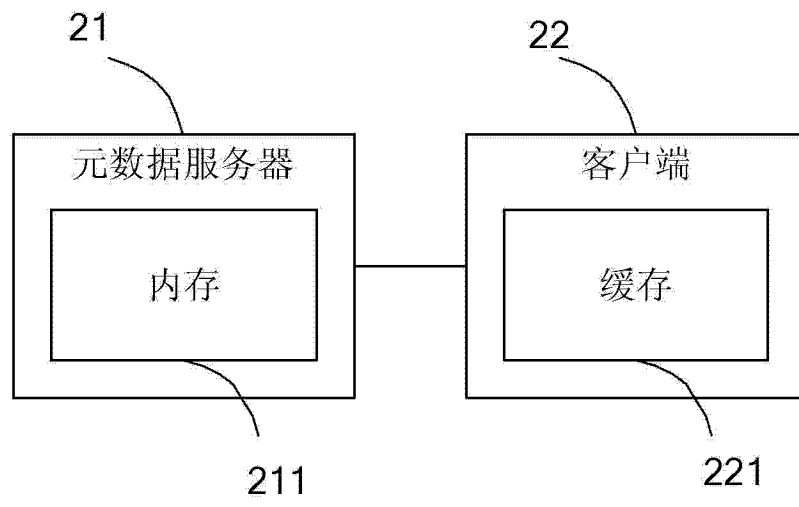


图 2