



(12) 发明专利申请

(10) 申请公布号 CN 113867628 A

(43) 申请公布日 2021. 12. 31

(21) 申请号 202111028315.3

(22) 申请日 2021.09.02

(71) 申请人 网易(杭州)网络有限公司

地址 310052 浙江省杭州市滨江区长河街
道网商路599号4幢7层

(72) 发明人 余利华 蒋鸿翔 范欣欣 章锡平

(74) 专利代理机构 北京博思佳知识产权代理有
限公司 11415

代理人 林祥

(51) Int. Cl.

G06F 3/06 (2006.01)

G06F 21/31 (2013.01)

G06F 21/62 (2013.01)

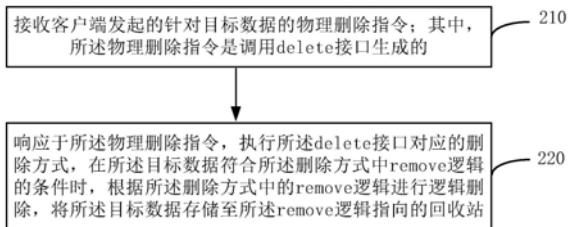
权利要求书2页 说明书18页 附图6页

(54) 发明名称

数据删除方法、装置、存储介质和计算设备

(57) 摘要

本公开的实施方式提供了一种数据删除方法、装置、存储介质和计算设备。包括:接收客户端发起的针对目标数据的物理删除指令;其中,所述物理删除指令是调用delete接口生成的;响应于所述物理删除指令,执行所述delete接口对应的删除方式,在所述目标数据符合所述删除方式中remove逻辑的条件时,根据所述删除方式中的remove逻辑进行逻辑删除,将所述目标数据存储至所述remove逻辑指向的回收站。本公开的方法使得在误删数据后能够及时发现恢复,且不影响分布式文件系统的服务稳定性。



1. 一种数据删除方法,应用于分布式文件系统中的管理节点,所述方法包括:
接收客户端发起的针对目标数据的物理删除指令;其中,所述物理删除指令是调用delete接口生成的;
响应于所述物理删除指令,执行所述delete接口对应的删除方式,在所述目标数据符合所述删除方式中remove逻辑的条件时,根据所述删除方式中的remove逻辑进行逻辑删除,将所述目标数据存储至所述remove逻辑指向的回收站。
2. 根据权利要求1所述的方法,所述执行所述delete接口对应的删除方式,包括:
确定所述目标数据的数据类型是否是目录类型;
响应于所述目标数据的数据类型不是目录类型,确定所述目标数据符合所述删除方式中remove逻辑的条件。
3. 根据权利要求2所述的方法,还包括:
响应于所述目标数据的数据类型是目录类型,确定所述物理删除指令的指令类型是否是递归删除;
响应于所述物理删除指令的指令类型是递归删除,确定所述目标数据符合所述删除方式中remove逻辑的条件。
4. 根据权利要求3所述的方法,还包括:
响应于所述目标数据的数据类型是目录类型,且所述物理删除指令的指令类型不是递归删除,确定所述目标数据是否为空;
响应于所述目标数据为空,确定所述目标数据符合所述删除方式中remove逻辑的条件。
5. 根据权利要求1所述的方法,所述执行所述delete接口对应的删除方式,包括:
确定所述目标数据是否是临时数据;其中,所述临时数据是从与所述分布式文件系统关联的上层应用系统获取的;
响应于所述目标数据不是临时数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。
6. 一种数据删除方法,应用于与分布式文件系统对应的客户端,所述方法包括:
拦截针对目标数据的物理删除指令,所述物理删除指令是调用delete接口生成的;
确定所述目标数据是否是临时数据;其中,所述临时数据是从与所述分布式文件系统关联的上层应用系统获取的;
响应于所述目标数据不是临时数据,调用remove接口以向分布式文件系统发起针对所述目标数据的逻辑删除指令;以使所述分布式文件系统中的管理节点响应于所述逻辑删除指令,执行所述remove接口对应的删除方式,将所述目标数据存储至所述分布式文件系统提供的回收站。
7. 一种数据删除装置,应用于分布式文件系统中的管理节点,所述装置包括:
接收单元,接收客户端发起针对目标数据的物理删除指令;其中,所述物理删除指令是调用delete接口生成的;
删除单元,响应于所述物理删除指令,执行所述delete接口对应的删除方式,在所述目标数据符合所述删除方式中remove逻辑的条件时,根据所述删除逻辑中的remove逻辑进行逻辑删除,将所述目标数据存储至所述remove逻辑指向的回收站。

8. 一种数据删除装置,应用于与分布式文件系统对应的客户端,所述装置包括:
拦截单元,拦截针对目标数据的物理删除指令,所述物理删除指令是调用delete接口生成的;

确定单元,确定所述目标数据是否是非临时数据;

删除单元,响应于所述目标数据为非临时数据时,调用remove接口以向分布式文件系统发起针对所述目标数据的逻辑删除指令;以使所述分布式文件系统中的管理节点响应于所述逻辑删除指令,执行所述remove接口对应的删除方式,将所述目标数据存储至所述分布式文件系统提供的回收站。

9. 一种计算机可读存储介质,包括:

当所述计算机可读存储介质中的指令由电子设备的处理器执行时,使得电子设备能够执行如权利要求1-6中任一项所述的数据删除方法。

10. 一种计算设备,包括:

处理器;

用于存储所述处理器可执行指令的存储器;

其中,所述处理器被配置为执行所述可执行指令,以实现如权利要求1-6中任一项所述的数据删除方法。

数据删除方法、装置、存储介质和计算设备

技术领域

[0001] 本公开的实施方式涉及计算机技术领域,更具体地,本公开的实施方式涉及一种数据删除方法、装置、存储介质和计算设备。

背景技术

[0002] 本部分旨在为权利要求书中陈述的本公开的实施方式提供背景或上下文。此处的描述不因为包括在本部分中就承认是现有技术。

[0003] HDFS (Hadoop Distributed File System,) 是一种被设计成适合运行在通用硬件 (Commodity Hardware) 上的分布式文件系统。在HDFS上可以存储海量的数据,并且随时都有针对这些数据的操作发生。

[0004] 对数据的删除操作是一种比较常规的操作。通常,删除操作有两种触发方式,一种是调用remove接口对数据进行逻辑删除;另一种是调用delete接口对数据进行物理删除。其中,逻辑删除是指将被删除数据会存放到HDFS上配置的回收站,在用户发现误删后可以基于回收站恢复误删的数据,以起到误删补救。而物理删除是指不经过回收站,直接删除数据,物理删除后的数据无法恢复。

发明内容

[0005] 但是,如果用户错误调用delete接口对数据进行删除,在相关技术中,还无法针对调用delete接口进行物理删除的数据进行恢复的措施。

[0006] 为此,非常需要一种改进的方案,针对被delete接口删除的数据进行恢复。

[0007] 在本公开实施方式的第一方面中,提供了一种数据删除方法,应用于分布式文件系统中的管理节点,包括:

[0008] 接收客户端发起的针对目标数据的物理删除指令;其中,所述物理删除指令是调用delete接口生成的;

[0009] 响应于所述物理删除指令,执行所述delete接口对应的删除方式,在所述目标数据符合所述删除方式中remove逻辑的条件时,根据所述删除方式中的remove逻辑进行逻辑删除,将所述目标数据存储至所述remove逻辑指向的回收站。

[0010] 可选的,所述执行所述delete接口对应的删除方式,包括:

[0011] 确定所述目标数据的数据类型是否是目录类型;

[0012] 响应于所述目标数据的数据类型不是目录类型,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0013] 可选的,还包括:

[0014] 响应于所述目标数据的数据类型是目录类型,确定所述物理删除指令的指令类型是否是递归删除;

[0015] 响应于所述物理删除指令的指令类型是递归删除,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0016] 可选的,还包括:

[0017] 响应于所述目标数据的数据类型是目录类型,且所述物理删除指令的指令类型不是递归删除,确定所述目标数据是否为空;

[0018] 响应于所述目标数据为空,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0019] 可选的,所述执行所述delete接口对应的删除方式,包括:

[0020] 确定所述目标数据是否是临时数据;其中,所述临时数据是从与所述分布式文件系统关联的上层应用系统获取的;

[0021] 响应于所述目标数据不是临时数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0022] 可选的,还包括:

[0023] 响应于所述目标数据是临时数据,根据所述删除方式中的delete逻辑将所述目标数据进行物理删除。

[0024] 可选的,所述确定所述目标数据是否是临时数据,包括:

[0025] 从与所述分布式文件系统关联的上层应用系统获取临时工作路径的列表;其中,所述临时工作路径为所述上层应用系统运行过程中产生的临时数据的存储路径;

[0026] 响应于所述临时工作路径的列表中记录有所述目标数据的存储路径,确定所述目标数据是临时数据;

[0027] 响应于所述临时工作路径的列表中未记录所述目标数据的存储路径,确定所述目标数据不是临时数据。

[0028] 可选的,所述执行所述delete接口对应的删除方式,包括:

[0029] 确定所述回收站中是否存储有与所述目标数据相同的已回收数据;

[0030] 响应于所述回收站中没有存储与所述目标数据相同的已回收数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0031] 可选的,还包括:

[0032] 响应于所述回收站中存储有与所述目标数据相同的已回收数据,在所述回收站中创建带时间戳的子目录,以及为所述子目录关联所述回收站中与所述目标数据相同的已回收数据。

[0033] 可选的,所述回收站包括分布式文件系统分配给超级用户的公共回收站;其中,在所述公共回收站的主目录下存储有每个客户端创建的对应的用户目录,所述用户目录用于存储对应于该用户目录的客户端所删除的数据。

[0034] 可选的,还包括:

[0035] 所述管理节点周期性地触发所述公共回收站配置的checkpoint函数,以物理删除所述公共回收站内存储时长超过预设时长的数据。

[0036] 可选的,所述将所述目标数据存储至所述remove逻辑中指向的回收站,包括:

[0037] 执行rename接口,将所述目标数据的存储路径修改为所述公共回收站的目录下所述客户端对应的用户目录。

[0038] 可选的,所述公共回收站的目录结构为三级目录结构;

[0039] 第一级目录为/user/hdfs/.Trash;

[0040] 第一级目录下的第二级目录为/user/hdfs/.Trash/Current;

[0041] 第二级目录下的第三级目录为/user/hdfs/.Trash/Current/\${USER};其中,\$ {USER}为不同客户端创建的不同用户目录;

[0042] 所述第一级目录和第二级目录基于所述分布式文件系统的权限管理模块向所有客户端开放的访问权限,所述第三级目录下的不同用户目录仅对唯一对应的客户端开放访问权限。

[0043] 在本公开实施方式的第二方面中,提供了一种数据删除方法,应用于与分布式文件系统对应的客户端,包括:

[0044] 拦截针对目标数据的物理删除指令,所述物理删除指令是调用delete接口生成的;

[0045] 确定所述目标数据是否是临时数据;其中,所述临时数据是从与所述分布式文件系统关联的上层应用系统获取的;

[0046] 响应于所述目标数据不是临时数据,调用remove接口以向分布式文件系统发起针对所述目标数据的逻辑删除指令;以使所述分布式文件系统中的管理节点响应于所述逻辑删除指令,执行所述remove接口对应的删除方式,将所述目标数据存储至所述分布式文件系统提供的回收站。

[0047] 可选的,还包括:

[0048] 响应于所述目标数据是临时数据,将所述物理删除指令发送给所述分布式文件系统;以使所述分布式文件系统中的管理节点响应于所述物理删除指令,执行所述delete接口对应的删除方式,将所述目标数据进行物理删除。

[0049] 可选的,所述确定所述目标数据是否是临时数据,包括:

[0050] 从与所述分布式文件系统关联的上层应用系统获取临时工作路径的列表;其中,所述临时工作路径为所述上层应用系统运行过程中产生的临时数据的存储路径;

[0051] 响应于所述临时工作路径的列表中记录有所述目标数据的存储路径,确定所述目标数据是临时数据;

[0052] 响应于所述临时工作路径的列表中未记录有所述目标数据的存储路径,确定所述目标数据不是临时数据。

[0053] 在本公开实施方式的第三方面中,提供了一种数据删除装置,应用于分布式文件系统中的管理节点,所述装置包括:

[0054] 接收单元,接收客户端发起针对目标数据的物理删除指令;其中,所述物理删除指令是调用delete接口生成的;

[0055] 删除单元,响应于所述物理删除指令,执行所述delete接口对应的删除方式,在所述目标数据符合所述删除方式中remove逻辑的条件时,根据所述删除逻辑中的remove逻辑进行逻辑删除,将所述目标数据存储至所述remove逻辑指向的回收站。

[0056] 可选的,所述删除单元,还包括:

[0057] 第一确定子单元,确定所述目标数据的数据类型是否是目录类型;响应于所述目标数据的数据类型不是目录类型,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0058] 可选的,所述删除单元,还包括:

[0059] 第二确定子单元,响应于所述目标数据的数据类型是目录类型,确定所述物理删除指令的指令类型是否是递归删除;响应于所述物理删除指令的指令类型是递归删除,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0060] 可选的,所述删除单元,还包括:

[0061] 第三确定子单元,响应于所述目标数据的数据类型是目录类型,且所述物理删除指令的指令类型不是递归删除,确定所述目标数据是否为空;响应于所述目标数据为空,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0062] 可选的,所述删除单元,还包括:

[0063] 第四确定子单元,确定所述目标数据是否是临时数据;其中,所述临时数据是从与所述分布式文件系统关联的上层应用系统获取的;响应于所述目标数据不是临时数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0064] 可选的,所述删除单元,还包括:

[0065] 第五确定子单元,响应于所述目标数据是临时数据,根据所述删除方式中的delete逻辑将所述目标数据进行物理删除。

[0066] 可选的,所述删除单元,还包括:

[0067] 获取子单元,从与所述分布式文件系统关联的上层应用系统获取临时工作路径的列表;其中,所述临时工作路径为所述上层应用系统运行过程中产生的临时数据的存储路径;

[0068] 第六确定子单元,响应于所述临时工作路径的列表中记录有所述目标数据的存储路径,确定所述目标数据是临时数据;响应于所述临时工作路径的列表中未记录所述目标数据的存储路径,确定所述目标数据不是临时数据。

[0069] 可选的,所述删除单元,还包括:

[0070] 第七确定子单元,确定所述回收站中是否存储有与所述目标数据相同的已回收数据;响应于所述回收站中没有存储与所述目标数据相同的已回收数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0071] 可选的,所述第七确定子单元,还包括:

[0072] 响应于所述回收站中存储有与所述目标数据相同的已回收数据,在所述回收站中创建带时间戳的子目录,以及为所述子目录关联所述回收站中与所述目标数据相同的已回收数据。

[0073] 可选的,所述回收站包括分布式文件系统分配给超级用户的公共回收站;其中,在所述公共回收站的主目录下存储有每个客户端创建的对应的用户目录,所述用户目录用于存储对应于该用户目录的客户端所删除的数据。

[0074] 可选的,所述装置还包括:

[0075] 清理单元,所述管理节点周期性地触发所述公共回收站配置的checkpoint函数,以物理删除所述公共回收站内存储时长超过预设时长的数据。

[0076] 可选的,所述删除单元,还包括:

[0077] 执行rename接口,将所述目标数据的存储路径修改为所述公共回收站的目录下所述客户端对应的用户目录。

[0078] 可选的,所述公共回收站的目录结构为三级目录结构;

- [0079] 第一级目录为/user/hdfs/.Trash;
- [0080] 第一级目录下的第二级目录为/user/hdfs/.Trash/Current;
- [0081] 第二级目录下的第三级目录为/user/hdfs/.Trash/Current/{USER};其中,{USER}为不同客户端创建的不同用户目录;
- [0082] 所述第一级目录和第二级目录基于所述分布式文件系统的权限管理模块向所有客户端开放的访问权限,所述第三级目录下的不同用户目录仅对唯一对应的客户端开放访问权限。
- [0083] 在本公开实施方式的第四方面中,提供了一种数据删除装置,应用于与分布式文件系统对应的客户端,所述装置包括:
- [0084] 拦截单元,拦截针对目标数据的物理删除指令,所述物理删除指令是调用delete接口生成的;
- [0085] 确定单元,确定所述目标数据是否是非临时数据;
- [0086] 删除单元,响应于所述目标数据为非临时数据时,调用remove接口以向分布式文件系统发起针对所述目标数据的逻辑删除指令;以使所述分布式文件系统中的管理节点响应于所述逻辑删除指令,执行所述remove接口对应的删除方式,将所述目标数据存储至所述分布式文件系统提供的回收站。
- [0087] 可选的,所述删除单元还包括:
- [0088] 响应于所述目标数据是临时数据,将所述物理删除指令发送给所述分布式文件系统;以使所述分布式文件系统中的管理节点响应于所述物理删除指令,执行所述delete接口对应的删除方式,将所述目标数据进行物理删除。
- [0089] 可选的,所述确定单元,包括:
- [0090] 获取子单元,从与所述分布式文件系统关联的上层应用系统获取临时工作路径的列表;其中,所述临时工作路径为所述上层应用系统运行过程中产生的临时数据的存储路径;
- [0091] 确定子单元,响应于所述临时工作路径的列表中记录有所述目标数据的存储路径,确定所述目标数据是临时数据;响应于所述临时工作路径的列表中未记录有所述目标数据的存储路径,确定所述目标数据不是临时数据。
- [0092] 在本公开实施方式的第五方面中,提供了一种计算机可读存储介质,包括:
- [0093] 当所述计算机可读存储介质中的指令由电子设备的处理器执行时,使得电子设备能够执行如前一项所述的数据删除方法。
- [0094] 在本公开实施方式的第六方面中,提供了一种计算设备,包括:
- [0095] 处理器;
- [0096] 用于存储所述处理器可执行指令的存储器;
- [0097] 其中,所述处理器被配置为执行所述可执行指令,以实现如前一项所述的数据删除方法。
- [0098] 根据本公开实施方式提供的数据删除方案,一方面,通过对分布式文件系统中的delete接口进行改进,在delete接口对应的删除方式中增加remove逻辑的判断条件,以将符合remove逻辑的条件的物理删除转换为逻辑删除。如此,当客户端调用delete接口对数据进行物理删除的情况下,分布式文件系统中管理节点也可以将物理删除转换为逻辑删

除,由于逻辑删除的数据可以被恢复,因此可以解决分布式文件系统中的管理节点响应物理删除指令进行删除的数据无法被恢复的问题。

[0099] 另一方面,通过对客户端中的调用delete接口的流程进行改进,在确定目标数据不是临时数据情况下,并不基于delete接口直接发起针对目标数据的物理删除指令,而是转而调用remove接口以发起针对目标数据的逻辑删除指令。如此,客户端即使调用delete接口对数据进行物理删除,也会转为向分布式文件系统发送逻辑删除指令,从而使得分布式文件系统中的管理节点响应于所述逻辑删除指令对数据进行逻辑删除。由于逻辑删除的数据可以被恢复,因此可以解决客户端调用delete接口进行删除的数据无法被恢复的问题。

附图说明

[0100] 通过参考附图阅读下文的详细描述,本公开示例性实施方式的上述以及其他目的、特征和优点将变得易于理解。在附图中,以示例性而非限制性的方式示出了本公开的若干实施方式,其中:

[0101] 图1示意性地示出了本公开提供的HDFS的系统示意图;

[0102] 图2示意性地示出了现有HDFS中remove接口删除数据的流程示意图;

[0103] 图3示意性地示出了现有HDFS中delete接口删除数据的流程示意图;

[0104] 图4示意性地示出了本公开提供的数据删除方法示意图;

[0105] 图5示意性地示出了本公开提供的公共回收站的目录结构示意图;

[0106] 图6示意性地示出了本公开提供的改进管理节点流程的delete接口删除数据的示意图;

[0107] 图7示意性地示出了本公开提供的数据删除方法示意图;

[0108] 图8示意性地示出了本公开提供的改进客户端流程的delete接口删除数据的示意图;

[0109] 图9示意性地示出了本公开提供的介质示意图;

[0110] 图10示意性地示出了本公开提供的数据删除装置示意图;

[0111] 图11示意性地示出了本公开提供的数据删除装置示意图;

[0112] 图12示意性地示出了本公开提供的计算设备示意图。

[0113] 在附图中,相同或对应的标号表示相同或对应的部分。

具体实施方式

[0114] 下面将参考若干示例性实施方式来描述本公开的原理和精神。应当理解,给出这些实施方式仅仅是为了使本领域技术人员能够更好地理解进而实现本公开,而并非以任何方式限制本公开的范围。相反,提供这些实施方式是为了使本公开更加透彻和完整,并且能够将本公开的范围完整地传达给本领域的技术人员。

[0115] 本领域技术人员知道,本公开的实施方式可以实现为一种系统、装置、设备、方法或计算机程序产品。因此,本公开可以具体实现为以下形式,即:完全的硬件、完全的软件(包括固件、驻留软件、微代码等),或者硬件和软件结合的形式。

[0116] 根据本公开的实施方式,提出了一种数据删除方法、计算机可读存储介质、装置和

计算设备。

[0117] 在本文中,需要理解的是,附图中的任何元素数量均用于示例而非限制,以及任何命名都仅用于区分,而不具有任何限制含义。

[0118] 下面参考本公开的若干代表性实施方式,详细阐释本公开的原理和精神。

[0119] 发明概述

[0120] 本公开人发现,如果用户错误调用delete接口对数据进行删除,在相关技术中,还无法针对调用delete接口进行物理删除的数据进行恢复的措施。

[0121] 在介绍了本公开的基本原理之后,下面具体介绍本公开的各种非限制性实施方式。

[0122] 应用场景总览

[0123] 首先参考图1所示的一种HDFS的系统架构图。在该系统架构图中,分布式文件系统可以包括若干数据节点(Data Node)、一个或多个管理节点(Name Node)。

[0124] 其中,数据节点用于存储业务相关的数据,例如用户账户可以存储在数据节点,用户使用某个业务后产生的业务数据也会存储在数据节点。

[0125] 管理节点用于对外提供服务,其维护了整个分布式文件系统的文件目录树,保存着文件数据块的地址。

[0126] 一般的,为了保证分布式文件系统的高可用性,可以部署两个或两个以上的管理节点。其中一个称为主管理节点,其它的称为备用管理节点。通常情况下,主管理节点用于对外提供服务,一旦主管理节点出现异常例如宕机,则备用管理节点可以快速成为主管理节点以对外提供服务。

[0127] 图1中还包括与分布式文件系统对应的客户端(Client),客户端可以是指安装有分布式文件系统应用程序的PC电脑、笔记本电脑、智能手机、智能平板、服务器等设备。

[0128] 客户端可以通过分布式系统对外提供的调用接口(API)向分布式文件系统中管理节点发起业务指令。以使管理节点响应业务指令,执行相关的业务流程,返回对应的业务结果等。其中,相关的业务结果可以存储到前述的数据节点。

[0129] 其中,所述业务指令可以包括新增数据、修改数据、删除数据等各种操作指令。一般的不同业务指令需要调用不同的业务接口才能产生。

[0130] 例如,客户端可以调用delete接口发起物理删除指令;或者可以调用remove接口发起逻辑删除指令。

[0131] 请参考图2所示现有HDFS中的remove接口删除数据的流程示意图。

[0132] 客户端调用remove接口生成并发送针对目标数据的逻辑删除指令;管理节点在接收到该逻辑删除指令后,响应于该逻辑删除指令,执行remove接口对应的逻辑删除方式,将目标数据存储至分布式文件系统提供的回收站。

[0133] 在分布式文件系统中,针对调用remove接口删除的数据,会将用户删除的数据暂时存放到该用户对应的回收站中。每个用户都对应有自己的回收站,回收站一般位于数据节点中(为了后续区分将现有分布式文件系统提供的回收站称为用户回收站)。

[0134] 分布式文件系统还设置有回收站清理程序,由管理节点周期性触发checkpoint函数以调用回收站清理程序清理用户回收站内的数据。

[0135] 请参考图3所示现有HDFS中的delete接口删除数据的流程示意图。客户端调用

delete接口生成并发送针对目标数据的物理删除指令；管理节点在接收到该物理删除指令后，响应于该物理删除指令，执行delete接口对应的物理删除方式，对目标数据进行物理删除。物理删除是指不经过回收站，直接删除数据，物理删除后的数据无法恢复。

[0136] 示例性方法

[0137] 下面结合图1所示的应用场景，参考图4和图6来描述根据本公开示例性实施方式的数据删除的方法。需要注意的是，上述应用场景仅是为了便于理解本公开的精神和原理而示出，本公开的实施方式在此方面不受任何限制。相反，本公开的实施方式可以应用于适用的任何场景。

[0138] 如图4所示，所述数据删除方法，可以应用于前述的分布式文件系统中的管理节点，并可以包括以下步骤：

[0139] 步骤210：接收客户端发起的针对目标数据的物理删除指令；其中，所述物理删除指令是调用delete接口生成的；

[0140] 步骤220：响应于所述物理删除指令，执行所述delete接口对应的删除方式，在所述目标数据符合所述删除方式中remove逻辑的条件时，根据所述删除方式中的remove逻辑进行逻辑删除，将所述目标数据存储至所述remove逻辑指向的回收站。

[0141] 本实施例中，需要对分布式文件系统中的delete接口进行改进。将delete接口对应的底层代码中增加一段进行逻辑删除的分支代码。

[0142] 具体地，通过确定所述目标数据是否符合所述删除方式中remove逻辑的条件，如果符合则执行上述增加的逻辑删除的分支代码，以实现目标数据的逻辑删除；如果不符合则执行delete接口对应的底层代码中原本的物理删除代码，以实现目标数据的物理删除。

[0143] 其中，上述remove逻辑的条件可以有多种条件，这些条件可以单独使用也可以任意组合使用，以下通过若干示例对其进行说明。

[0144] 在一示例性的实施例中，上述执行所述delete接口对应的删除方式，可以包括：

[0145] 确定所述目标数据的数据类型是否是目录类型；

[0146] 响应于所述目标数据的数据类型不是目录类型，确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0147] 该示例中，可以以数据类型作为remove逻辑的条件。在分布式文件系统的文件目录树中，如果用户删除的数据不是一个文件目录，则意味着被删除数据的路径位于文件目录树的叶子节点，由于不存在子目录，则不存在异常删除。本次删除操作是正常删除，符合删除方式中remove逻辑的条件。

[0148] 在一示例性的实施例中，在上述确定所述目标数据的数据类型是否是目录类型之后，还可以包括：

[0149] 响应于所述目标数据的数据类型是目录类型，确定所述物理删除指令的指令类型是否是递归删除；

[0150] 响应于所述物理删除指令的指令类型是递归删除，确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0151] 该示例中，针对数据类型是目录类型的情况，还可以以递归删除作为remove逻辑的条件。在分布式文件系统的文件目录树中，如果用户删除的数据是一个文件目录，由于文

件目录下可能存在各种子目录,因此需要确定本次删除是否是递归删除。其中,递归删除是指删除当前目录,以及当前目录下所有子目录的数据。

[0152] 由于递归删除可以删除当前目录以及其所有子目录的数据,因此不存在异常删除,本次删除操作是正常删除,符合删除方式中remove逻辑的条件。

[0153] 在一示例性的实施例中,在上述确定所述物理删除指令的指令类型是否是递归删除之后,还可以包括:

[0154] 响应于所述目标数据的数据类型是目录类型,且所述物理删除指令的指令类型不是递归删除,确定所述目标数据是否为空;

[0155] 响应于所述目标数据为空,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0156] 该示例中,针对数据类型是目录类型且不是递归删除的情况,还可以以目标数据是否为空作为remove逻辑的条件。在分布式文件系统的文件目录树中,如果用户删除的数据是一个文件目录,并且该文件目录下为空,则不存在异常删除,本次删除操作是正常删除,符合删除方式中remove逻辑的条件。

[0157] 而如果所述目标数据不为空,则说明本次针对目标数据的物理删除属于异常删除,不响应该物理删除指令。

[0158] 通过上述示例可以排除异常删除,放行正常删除的物理删除指令进一步做以下处理。

[0159] 在一示例性的实施例中,上述执行所述delete接口对应的删除方式,包括:

[0160] 确定所述目标数据是否是临时数据;其中,所述临时数据是从与所述分布式文件系统关联的上层应用系统获取的;

[0161] 响应于所述目标数据不是临时数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0162] 响应于所述目标数据是临时数据,根据所述删除方式中的delete逻辑将所述目标数据进行物理删除。

[0163] 该示例中,针对如何确定目标数据是否符合所述删除方式中remove逻辑的条件,还可以从目标数据是否是临时数据出发进行判断。

[0164] 其中,本示例中的临时数据可以是指与所述分布式文件系统关联的上层应用系统运行过程中产生的临时数据,例如上层应用系统运行任务的任务包、任务中间数据、配置数据、日志数据等。

[0165] 在实际应用中,与分布式文件系统关联的上层应用系统可以是指适配HDFS的大规模数据处理系统,如YARN(Yet Another Resource Negotiator)、SPARK等。

[0166] 这些上层应用系统在运行与业务相关的任务时会产生临时数据,这些临时数据对于用户来说通常是不被感知的,且与用户业务的相关度较低、甚至完全不相关。

[0167] 众所周知数据存储不管是逻辑存储还是物理存储都需要消耗一定的存储资源。

[0168] 对于本公开来说,由于需要将原本用户物理删除的数据转化逻辑删除,因此必然会占用现有的存储资源。而为了提高存储资源的利用率,或者减少无意义数据的存储,该示例中针对非临时数据的物理删除指令才需要转化为逻辑删除,而针对临时数据的物理删除指令依然沿用物理删除的删除逻辑直接进行物理删除。

[0169] 本公开中,为了解决前述调用delete接口删除的数据无法被恢复的问题,发明人在改进过程中基于碰到的各种技术障碍,通盘考虑并设置了前述的这些remove逻辑的触发条件。在解决调用delete接口删除的数据无法被恢复的问题过程中,严格遵循上述remove逻辑的触发机制,能够为数据的删除和留存提供合理依据,并对用户的误删操作提供了补救措施。

[0170] 在一示例性的实施例中,所述确定所述目标数据是否是临时数据,包括:

[0171] 从与所述分布式文件系统关联的上层应用系统获取临时工作路径的列表;其中,所述临时工作路径为所述上层应用系统运行过程中产生的临时数据的存储路径;

[0172] 响应于所述临时工作路径的列表中记录有所述目标数据的存储路径,确定所述目标数据是临时数据;

[0173] 响应于所述临时工作路径的列表中未记录所述目标数据的存储路径,确定所述目标数据不是临时数据。

[0174] 在该示例中,提供了一种从上层应用系统获取临时工作路径的列表,由于该列表中记录了上层应用系统运行过程中产生的临时数据的存储路径;因此只要比对目标数据的存储路径和该列表中的临时工作路径,就可以确定目标数据是否是临时数据。由于工作路径的信息量较小,因此采用这种方式比对效率较高,可以更为快速地确定目标数据是否为临时数据的结果。

[0175] 当然,在实际应用也可以采用其他方式确定目标数据是否是临时数据。例如将目标数据的存储路径发送到上层应用系统,由上层应用系统进行确定。再例如,管理节点还可以比较目标数据本身,即比较目标数据是否与上层应用系统的临时数据相同,如果相同则说明目标数据是临时数据。

[0176] 如前所述在分布式文件系统中,针对调用remove接口删除的数据,会将用户删除的数据暂时存放到该用户对应的回收站中。

[0177] 本实施例中,针对客户端调用delete接口发起的针对目标数据的物理删除指令,如果目标数据符合delete接口对应的删除方式中remove逻辑的条件;那么就会将原本需要物理删除的目标数据,转换为逻辑删除,从而将目标数据存放到remove逻辑指向的回收站。

[0178] 这里的回收站可以是指使用上述客户端发起物理删除指令的用户对应的用户回收站。

[0179] 在一示例性的实施例中,上述回收站还可以包括分布式文件系统分配给超级用户的公共回收站;其中,在所述公共回收站的主目录下存储有每个客户端创建的对应的用户目录,所述用户目录用于存储对应于该用户目录的客户端所删除的数据。

[0180] 其中,超级用户是分布式文件系统中一个技术概念,即启动分布式文件系统中管理节点的用户。

[0181] 该示例中,公共回收站实际上也是超级用户的用户回收站,也即分布式文件系统为超级用户设置的。

[0182] 但是,本实施例对超级用户的用户回收站进行了改造,使得超级用户的用户回收站可以被所有用户使用,而不是仅被超级用户使用。

[0183] 正是由于所有用户都可以使用,因此才将超级用户的用户回收站称为公共回收站。

[0184] 当然由于所有用户都会使用同一个回收站(即公共回收站),因而被不同用户所删除的数据就会混杂在一起进行存储,为此有必要解决由于数据混杂存储而导致的数据归属不清晰的问题。

[0185] 基于此,在本公开一示例性实施例中,在所述公共回收站的主目录下存储有每个客户端创建的对应的用户目录,所述用户目录用于存储对应于该用户目录的客户端所删除的数据。

[0186] 也就是说,通过建立各个用户对应的用户目录,虽然所有用户可以访问公共回收站的主目录,但是用户目录依然只有用户本人才能访问。如此,就可以将不同用户所删除的数据区分开来。

[0187] 在一示例性的实施例中,所述公共回收站的目录结构为三级目录结构;

[0188] 如图5所示,第一级目录为/user/hdfs/.Trash;

[0189] 第一级目录下的第二级目录为/user/hdfs/.Trash/Current;

[0190] 第二级目录下的第三级目录为/user/hdfs/.Trash/Current/\${USER};其中,\${USER}为不同客户端创建的不同用户目录;

[0191] 其中,所述第一级目录和第二级目录基于所述分布式文件系统的权限管理模块向所有客户端开放的访问权限,所述第三级目录下的不同用户目录仅对唯一对应的客户端开放访问权限。

[0192] 该示例中,通过将公共回收站的目录结构设置为三级目录结构,第一级目录设置为所有用户都可访问(可读可写),该路径为公共回收站(超级用户的用户回收站)的路径。在公共回收站初始化完成后,由于所有用户都可访问,因而保证了每一个访问公共回收站的用户都能够基于被赋予的可读可写权限创建这一级目录。

[0193] 第二级目录用于在系统自动触发checkpoint时,会重命名第二级目录的目录名,一般是将“Current”修改为“<date>”,然后通过checkpoint函数物理删除公共回收站内存储时长超过预设时长的数据。在第二级目录被重命名后,用户就无法访问第三级目录;因此需要为所有用户开放修改第二级目录的权限,以使每一个访问公共回收站的用户能够基于该修改第二级目录的权限将第二级目录的目录名修改回“Current”,这样用户就又可以访问第三级目录。如此,通过向用户开放修改第二级目录的权限,在兼顾了系统的checkpoint机制的同时,也确保了用户能够顺利的访问相应的第三级目录,从而方便用户对逻辑删除的数据进行恢复。

[0194] 第三级目录只对当前用户开放,用于隔离其他的用户。每个被用户通过delete接口删除但最终进行逻辑删除的数据都会存放到公共回收站的第三级目录中的用户目录中。如此,仅通过目录结构和目录名的规制,就实现了对不同用户的数据进行区管理,即不同用户所删除的数据都存放在各用户自有访问权限的各个用户目录中,从而避免了由于不同用户删除的数据混杂存储而导致的数据归属不清晰的问题。

[0195] 在一示例性的实施例中,所述将所述目标数据存储至所述remove逻辑中指向的回收站,包括:

[0196] 执行rename接口,将所述目标数据的存储路径修改为所述公共回收站的目录下所述客户端对应的用户目录。

[0197] 该示例中,通过rename接口将目标数据原本的存储路径修改为公共回收站的目录

下所述客户端对应的用户目录,是一种成本最低的数据移动方式。解决了将目标数据从原本的存储路径对应的内存空间移动到用户目录对应的内存空间所造成数据移存的低效率问题。

[0198] 在一示例性的实施例中,所述公共回收站同样设置有回收站清理程序,由管理节点周期性周期性地触发所述公共回收站配置的checkpoint函数,以物理删除所述公共回收站内存储时长超过预设时长的数据。

[0199] 其中,预设时长可以是预先设置的一个经验值,例如1个小时,则checkpoint函数被触发后,都会清理公共回收站中存储时长超过1小时的数据。

[0200] 通过回收站清理程序可以清理公共回收站中存储时间过长的数据,从而避免公共回收站的存储资源被耗尽,提高了存储资源的利用率。

[0201] 在一示例性的实施例中,所述执行所述delete接口对应的删除方式,包括:

[0202] 确定所述回收站中是否存储有与所述目标数据相同的已回收数据;

[0203] 响应于所述回收站中没有存储与所述目标数据相同的已回收数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0204] 在该示例中,回收站不会同时存储相同的数据,只有在回收站没有存储该删除的目标数据时,才会将目标数据存储到回收站。这样进一步提高了存储资源的利用率。

[0205] 需要说明的是,在回收站存储了与该删除的目标数据相同的已回收数据时,并不是说就不对目标数据进行逻辑删除了,而是不再将目标数据重复存储到回收站。需要提供另一种方式记录下这个被删除的目标数据。

[0206] 在一示例性的实施例中,响应于所述回收站中存储有与所述目标数据相同的已回收数据,在所述回收站中创建带时间戳的子目录,以及为所述子目录关联所述回收站中与所述目标数据相同的已回收数据。

[0207] 该示例中,虽然目标数据没有存储到回收站,但是由于回收站中已经有了相同的已回收数据,我们只要复用这个已回收数据即可记录本次被删除的目标数据。

[0208] 即在所述回收站中创建带时间戳的子目录,以及为所述子目录关联所述回收站中与所述目标数据相同的已回收数据。由于带时间戳的子目录是空的,因此在数据量上可以忽略不计,远远小于已回收数据的数据量,因此不仅可以记录下本次被删除的目标数据,通过时间戳还可以确定删除的时间(与已回收数据必然是不同的时间),而且通过关联关系也可以得到实际的数据(已回收数据)。如此不仅可以减少重复存储,以提高存储数据的利用率,而且也可以区分同一数据的不同的删除操作(时间戳)。需要说明的是,该示例中的回收站既可以指用户回收站,也可以是指公共回收站。通过该示例都可以减少回收站中的重复存储,提高回收站存储资源的利用效率。

[0209] 以下请参考图6所示本公开改进管理节点流程的delete接口删除数据的示意图;其对应了图4所示的方法实施例。将图6与图3进行比较可以发现:

[0210] 同样是客户端调用delete接口,生成并发送针对目标数据的物理删除指令,改进后HDFS的管理节点在响应物理删除指令后,并不是直接对目标数据进行物理删除,而是需要确定目标数据是否符合删除方式中remove逻辑的条件,并在目标数据符合删除方式中remove逻辑的条件时,根据删除方式中的remove逻辑进行逻辑删除,将目标数据存储至所述remove逻辑指向的回收站。

[0211] 通过上述实施例,通过对分布式文件系统中的delete接口进行改进,在delete接口对应的删除方式中增加remove逻辑的判断条件,以将符合remove逻辑的条件的物理删除转换为逻辑删除。如此,当客户端调用delete接口对数据进行物理删除的情况下,分布式文件系统中管理节点也可以将物理删除转换为逻辑删除,由于逻辑删除的数据可以被恢复,因此可以解决分布式文件系统中的管理节点响应物理删除指令进行删除的数据无法被恢复的问题。

[0212] 本公开除了提供上述对分布式文件系统中的delete接口进行改进的实施例之外,还提供了对客户端中调用delete接口进行改进的实施例。

[0213] 下面参考图7所示的数据删除方法,可以应用于与分布式文件系统对应的客户端,并可以包括以下步骤:

[0214] 步骤310:拦截针对目标数据的物理删除指令,所述物理删除指令是调用delete接口生成的;

[0215] 步骤320:确定所述目标数据是否是临时数据;其中,所述临时数据是从与所述分布式文件系统关联的上层应用系统获取的;

[0216] 步骤330:响应于所述目标数据不是临时数据,调用remove接口以向分布式文件系统发起针对所述目标数据的逻辑删除指令;以使所述分布式文件系统中的管理节点响应于所述逻辑删除指令,执行所述remove接口对应的删除方式,将所述目标数据存储至所述分布式文件系统提供的回收站。

[0217] 本实施例中,需要对客户端中调用delete接口后的流程进行改进。在确定目标数据不是临时数据时,不是基于delete接口以发起物理删除指令,而是转而调用remove接口以发起针对目标数据的逻辑删除指令。

[0218] 与前述图4实施例不同的是,该实施例中是对客户端侧进行改进,而图2实施例中是对分布式文件系统进行改进。该实施例中分布式文件系统的管理节点在接收到物理删除指令时进行物理删除,接收到逻辑删除指令时进行逻辑删除。

[0219] 在一示例性的实施例中,还包括:

[0220] 响应于所述目标数据是临时数据,将所述物理删除指令发送给所述分布式文件系统;以使所述分布式文件系统中的管理节点响应于所述物理删除指令,执行所述delete接口对应的删除方式,将所述目标数据进行物理删除。

[0221] 对于本公开来说,由于需要将原本用户物理删除的数据转化逻辑删除,因此必然会占用现有的存储资源。而为了提高存储资源的利用率,或者减少无意义数据的存储,该示例中针对非临时数据的物理删除指令才需要转化为逻辑删除,而针对临时数据的物理删除指令依然沿用物理删除的删除逻辑直接进行物理删除。

[0222] 与前述图4实施例类似的,所述确定所述目标数据是否是临时数据,包括:

[0223] 从与所述分布式文件系统关联的上层应用系统获取临时工作路径的列表;其中,所述临时工作路径为所述上层应用系统运行过程中产生的临时数据的存储路径;

[0224] 响应于所述临时工作路径的列表中记录有所述目标数据的存储路径,确定所述目标数据是临时数据;

[0225] 响应于所述临时工作路径的列表中未记录有所述目标数据的存储路径,确定所述目标数据不是临时数据。

[0226] 该示例的详细介绍可以参考前述实施例,此处不再进行赘述。

[0227] 以下请参考图8所示本公开改进客户端流程的delete接口删除数据的示意图;其对应了图7所示的方法实施例。将图8与图3进行比较可以发现:

[0228] 同样是客户端调用delete接口,生成针对目标数据的物理删除指令后,改进后客户端并不是直接就发送物理删除指令,而是会拦截该物理删除指令以确定目标数据是否是临时数据,并在目标数据不是临时数据时,调用remove接口向HDFS发起针对目标数据的逻辑删除指令。从而使得HDFS中的管理节点从图3执行物理删除变为图8执行逻辑删除,即响应于逻辑删除指令,执行remove接口对应的删除方式,将目标数据存储至分布式文件系统提供的回收站。

[0229] 通过上述实施例,通过对客户端中的调用delete接口的流程进行改进,在确定目标数据不是临时数据情况下,并不基于delete接口直接发起针对目标数据的物理删除指令,而是转而调用remove接口以发起针对目标数据的逻辑删除指令。如此,客户端即使调用delete接口对数据进行物理删除,也会转为向分布式文件系统发送逻辑删除指令,从而使得分布式文件系统中的管理节点响应于所述逻辑删除指令对数据进行逻辑删除。由于逻辑删除的数据可以被恢复,因此可以解决客户端调用delete接口进行删除的数据无法被恢复的问题。

[0230] 示例性介质

[0231] 在介绍了本公开示例性实施方式的方法之后,接下来,参考图9对本公开示例性实施方式的介质进行说明。

[0232] 本示例性实施方式中,可以通过程序产品实现上述方法,如可以采用便携式紧凑盘只读存储器(CD-ROM)并包括程序代码,并可以在设备,例如个人电脑上运行。然而,本公开的程序产品不限于此,在本文件中,可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。

[0233] 该程序产品可以采用一个或多个可读介质的任意组合。可读介质可以是可读信号介质或者可读存储介质。可读存储介质例如可以为但不限于电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。可读存储介质的更具体的例子(非穷举的列表)包括:具有一个或多个导线的电连接、便携式盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。

[0234] 计算机可读信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了可读程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。可读信号介质还可以是可读存储介质以外的任何可读介质,该可读介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。

[0235] 可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于无线、有线、光缆、RE等等,或者上述的任意合适的组合。

[0236] 可以以一种或多种程序设计语言的任意组合来编写用于执行本公开操作的程序代码,程序设计语言包括面向对象的程序设计语言,诸如Java、C++等,还包括常规的过程式程序设计语言,诸如C语言或类似的设计语言。程序代码可以完全地在用户计算设备上

执行、部分在用户计算设备上部分在远程计算设备上执行、或者完全在远程计算设备或服务服务器上执行。在涉及远程计算设备的情形中,远程计算设备可以通过任意种类的网络,包括局域网(LAN)或广域网(WAN),连接到用户计算设备,或者,可以连接到外部计算设备(例如利用因特网服务提供商来通过因特网连接)。

[0237] 示例性装置

[0238] 在介绍了本公开示例性实施方式的介质之后,接下来,参考图10、图11对本公开示例性实施方式的装置进行说明。

[0239] 图10示意性地示出了根据本公开实施方式的一种数据删除装置的框图,对应于前述图4所示的方法实施例。应用于分布式文件系统中的管理节点,该数据删除装置可以包括:

[0240] 接收单元510,接收客户端发起针对目标数据的物理删除指令;其中,所述物理删除指令是调用delete接口生成的;

[0241] 删除单元520,响应于所述物理删除指令,执行所述delete接口对应的删除方式,在所述目标数据符合所述删除方式中remove逻辑的条件时,根据所述删除逻辑中的remove逻辑进行逻辑删除,将所述目标数据存储至所述remove逻辑指向的回收站。

[0242] 可选的,所述删除单元520,还包括:

[0243] 第一确定子单元521,确定所述目标数据的数据类型是否是目录类型;响应于所述目标数据的数据类型不是目录类型,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0244] 可选的,所述删除单元520,还包括:

[0245] 第二确定子单元522,响应于所述目标数据的数据类型是目录类型,确定所述物理删除指令的指令类型是否是递归删除;响应于所述物理删除指令的指令类型是递归删除,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0246] 可选的,所述删除单元520,还包括:

[0247] 第三确定子单元523,响应于所述目标数据的数据类型是目录类型,且所述物理删除指令的指令类型不是递归删除,确定所述目标数据是否为空;响应于所述目标数据为空,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0248] 可选的,所述删除单元520,还包括:

[0249] 第四确定子单元524,确定所述目标数据是否是临时数据;其中,所述临时数据是从与所述分布式文件系统关联的上层应用系统获取的;响应于所述目标数据不是临时数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0250] 可选的,所述删除单元520,还包括:

[0251] 第五确定子单元525,响应于所述目标数据是临时数据,根据所述删除方式中的delete逻辑将所述目标数据进行物理删除。

[0252] 可选的,所述删除单元520,还包括:

[0253] 获取子单元526,从与所述分布式文件系统关联的上层应用系统获取临时工作路径的列表;其中,所述临时工作路径为所述上层应用系统运行过程中产生的临时数据的存储路径;

[0254] 第六确定子单元527,响应于所述临时工作路径的列表中记录有所述目标数据的

存储路径,确定所述目标数据是临时数据;响应于所述临时工作路径的列表中未记录所述目标数据的存储路径,确定所述目标数据不是临时数据。

[0255] 可选的,所述删除单元520,还包括:

[0256] 第七确定子单元,确定所述回收站中是否存储有与所述目标数据相同的已回收数据;响应于所述回收站中没有存储与所述目标数据相同的已回收数据,确定所述目标数据符合所述删除方式中remove逻辑的条件。

[0257] 可选的,第七确定子单元还包括:

[0258] 响应于所述回收站中存储有与所述目标数据相同的已回收数据,在所述回收站中创建带时间戳的子目录,以及为所述子目录关联所述回收站中与所述目标数据相同的已回收数据。

[0259] 可选的,所述回收站包括分布式文件系统分配给超级用户的公共回收站;其中,在所述公共回收站的主目录下存储有每个客户端创建的对应的用户目录,所述用户目录用于存储对应于该用户目录的客户端所删除的数据。

[0260] 可选的,所述装置还包括:

[0261] 清理单元530,所述管理节点周期性地触发所述公共回收站配置的checkpoint函数,以物理删除所述公共回收站内存储时长超过预设时长的数据。

[0262] 可选的,删除单元520,还包括:

[0263] 执行rename接口,将所述目标数据的存储路径修改为所述公共回收站的目录下所述客户端对应的用户目录。

[0264] 可选的,所述公共回收站的目录结构为三级目录结构;

[0265] 第一级目录为/user/hdfs/.Trash;

[0266] 第一级目录下的第二级目录为/user/hdfs/.Trash/Current;

[0267] 第二级目录下的第三级目录为/user/hdfs/.Trash/Current/\${USER};其中,\${USER}为不同客户端创建的不同用户目录;

[0268] 所述第一级目录和第二级目录基于所述分布式文件系统的权限管理模块向所有客户端开放的访问权限,所述第三级目录下的不同用户目录仅对唯一对应的客户端开放访问权限。

[0269] 图11示意性地示出了根据本公开实施方式的一种数据删除装置的框图,对应于前述图7所示的方法实施例。应用于与分布式文件系统对应的客户端,该数据删除装置可以包括:

[0270] 拦截单元610,拦截针对目标数据的物理删除指令,所述物理删除指令是调用delete接口生成的;

[0271] 确定单元620,确定所述目标数据是否是非临时数据;

[0272] 删除单元630,响应于所述目标数据为非临时数据时,调用remove接口以向分布式文件系统发起针对所述目标数据的逻辑删除指令;以使所述分布式文件系统中的管理节点响应于所述逻辑删除指令,执行所述remove接口对应的删除方式,将所述目标数据存储至所述分布式文件系统提供的回收站。

[0273] 可选的,删除单元630还包括:

[0274] 响应于所述目标数据是临时数据,将所述物理删除指令发送给所述分布式文件系

统;以使所述分布式文件系统中的管理节点响应于所述物理删除指令,执行所述delete接口对应的删除方式,将所述目标数据进行物理删除。

[0275] 可选的,所述确定单元620,包括:

[0276] 获取子单元621,从与所述分布式文件系统关联的上层应用系统获取临时工作路径的列表;其中,所述临时工作路径为所述上层应用系统运行过程中产生的临时数据的存储路径;

[0277] 确定子单元622,响应于所述临时工作路径的列表中记录有所述目标数据的存储路径,确定所述目标数据是临时数据;响应于所述临时工作路径的列表中未记录有所述目标数据的存储路径,确定所述目标数据不是临时数据。

[0278] 示例性计算设备

[0279] 在介绍了本公开示例性实施方式的方法、介质和装置之后,接下来,参考图12对本公开示例性实施方式的计算设备进行说明。

[0280] 图12显示的计算设备1500仅仅是一个示例,不应对本公开实施例的功能和使用范围带来任何限制。

[0281] 如图12所示,计算设备1500以通用计算设备的形式表现。计算设备1500的组件可以包括但不限于:上述至少一个处理单元1501、上述至少一个存储单元1502,连接不同系统组件(包括处理单元1501和存储单元1502)的总线1503。

[0282] 总线1503包括数据总线、控制总线和地址总线。

[0283] 存储单元1502可以包括易失性存储器形式的可读介质,例如随机存取存储器(RAM) 15021和/或高速缓存存储器15022,可以进一步包括非易失性存储器形式的可读介质,例如只读存储器(ROM) 15023。

[0284] 存储单元1502还可以包括具有一组(至少一个)程序模块15024的程序/实用工具15025,这样的程序模块15024包括但不限于:操作系统、一个或者多个应用程序、其它程序模块以及程序数据,这些示例中的每一个或某种组合中可能包括网络环境的实现。

[0285] 计算设备1500也可以与一个或多个外部设备1504(例如键盘、指向设备等)通信。

[0286] 这种通信可以通过输入/输出(I/O)接口1505进行。并且,计算设备1500还可以通过网络适配器1506与一个或者多个网络(例如局域网(LAN),广域网(WAN)和/或公共网络,例如因特网)通信。如图12所示,网络适配器1506通过总线1503与计算设备1500的其它模块通信。应当理解,尽管图中未示出,可以结合计算设备1500使用其它硬件和/或软件模块,包括但不限于:微代码、设备驱动器、冗余处理单元、外部磁盘驱动阵列、RAID系统、磁带驱动器以及数据备份存储系统等。

[0287] 应当注意,尽管在上文详细描述中提及了数据删除装置的若干单元/模块或子单元/模块,但是这种划分仅仅是示例性的并非强制性的。实际上,根据本公开的实施方式,上文描述的两个或更多单元/模块的特征和功能可以在一个单元/模块中具体化。反之,上文描述的一个单元/模块的特征和功能可以进一步划分为由多个单元/模块来具体化。

[0288] 此外,尽管在附图中以特定顺序描述了本公开方法的操作,但是,这并非要求或者暗示必须按照该特定顺序来执行这些操作,或是必须执行全部所示的操作才能实现期望的结果。附加地或备选地,可以省略某些步骤,将多个步骤合并为一个步骤执行,和/或将一个步骤分解为多个步骤执行。

[0289] 虽然已经参考若干具体实施方式描述了本公开的精神和原理,但是应该理解,本公开并不限于所公开的具体实施方式,对各方面的划分也不意味着这些方面中的特征不能组合以进行受益,这种划分仅是为了表述的方便。本公开旨在涵盖所附权利要求的精神和范围内所包括的各种修改和等同布置。

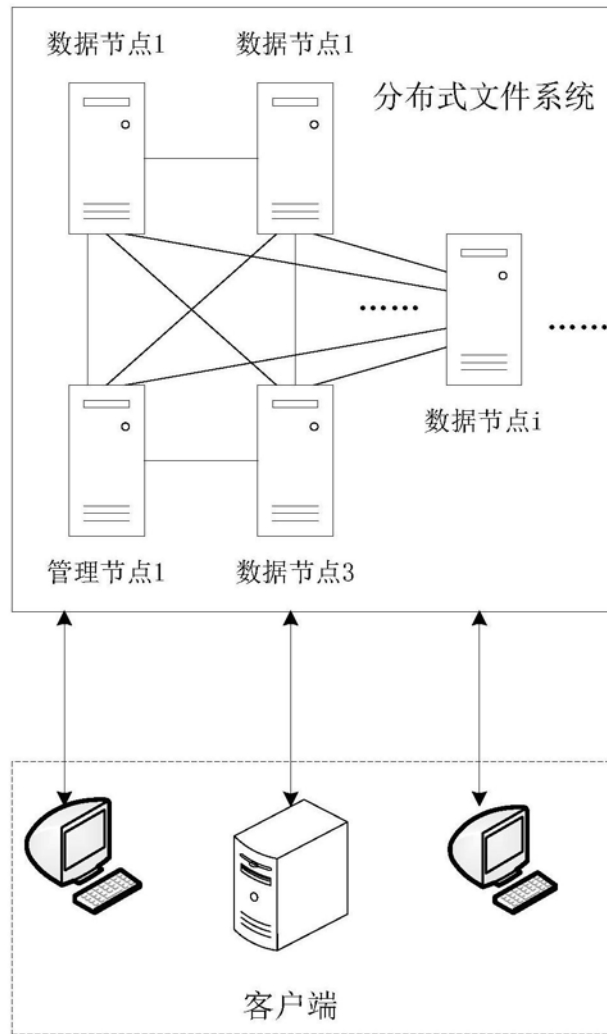


图1

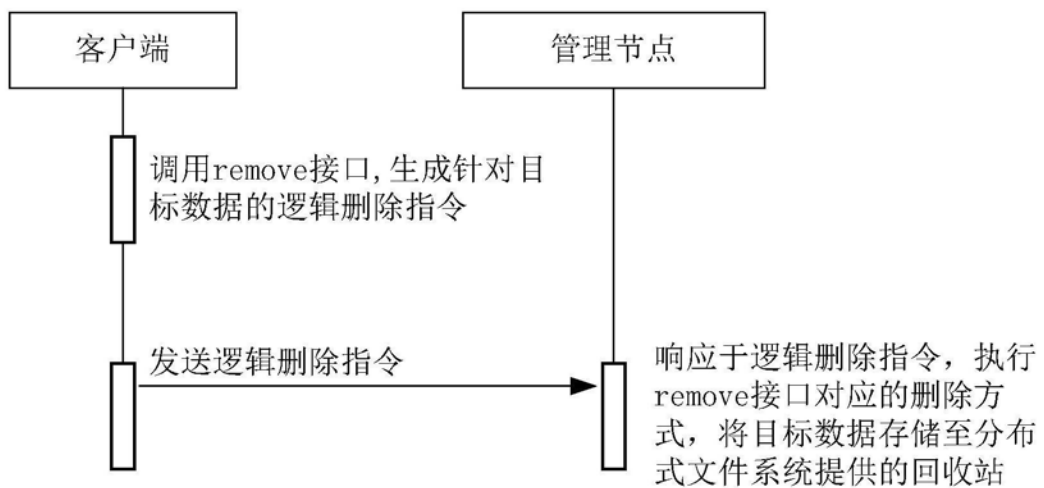


图2

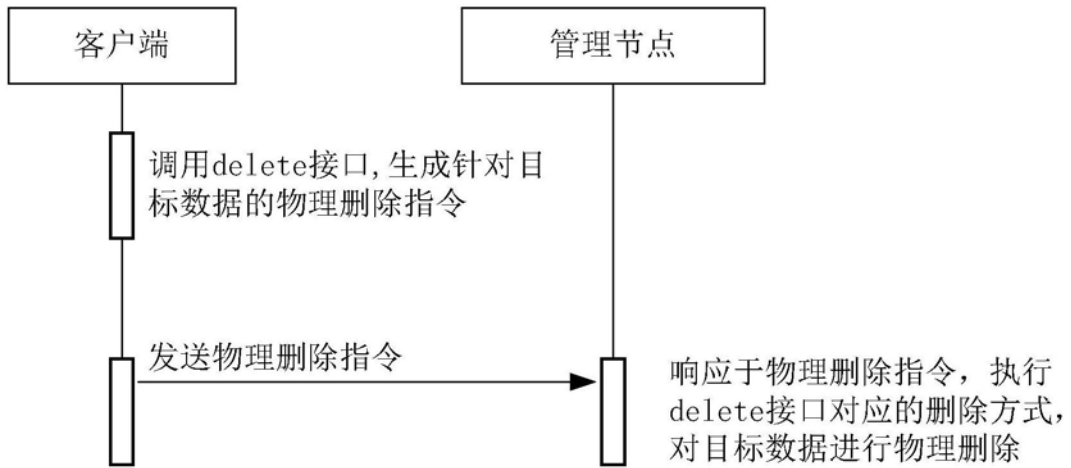


图3

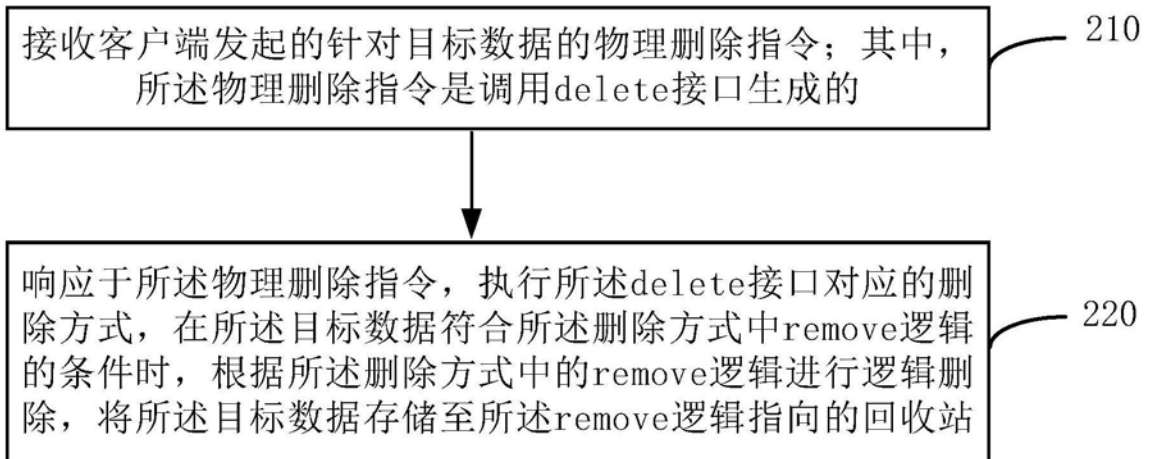


图4

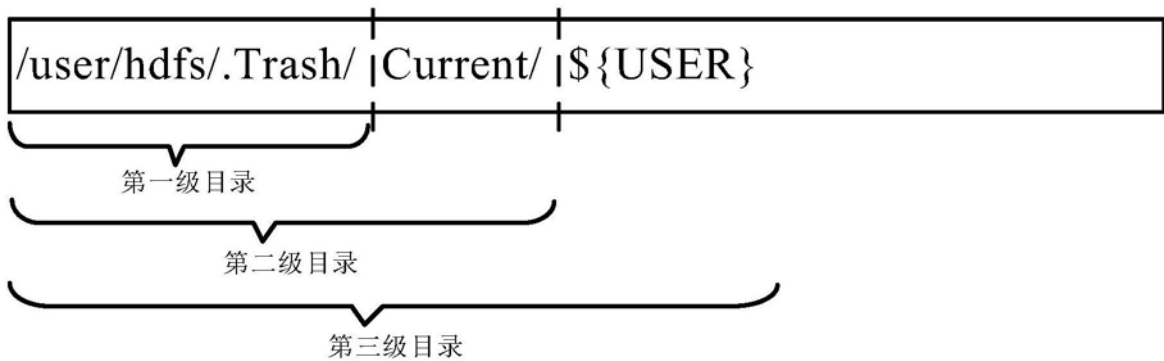


图5

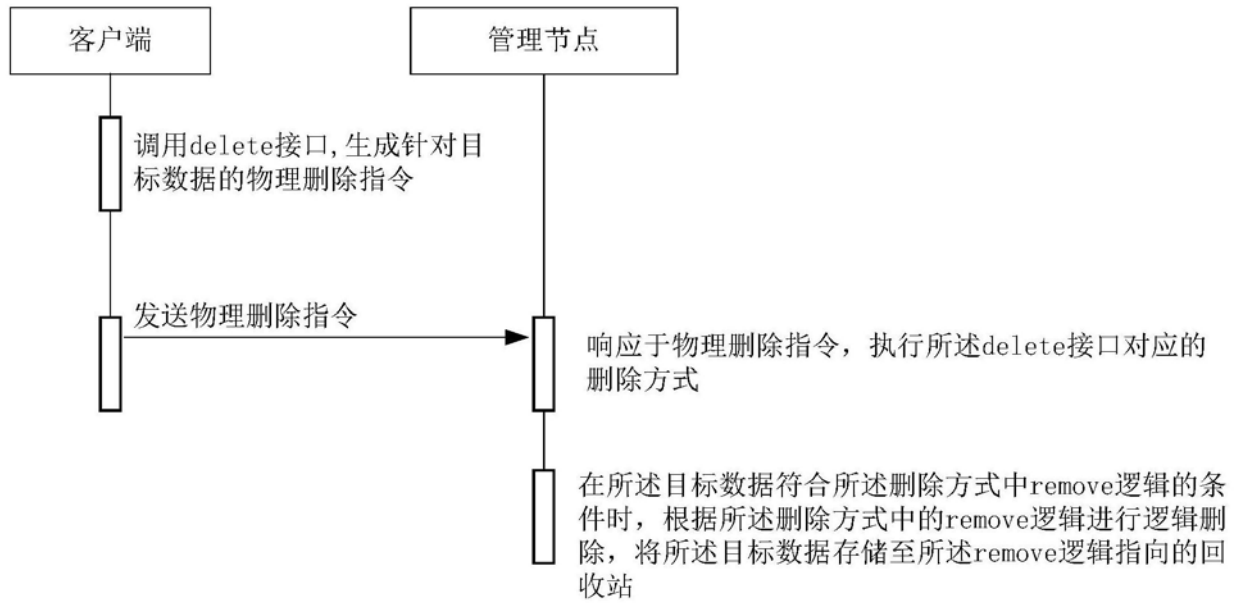


图6

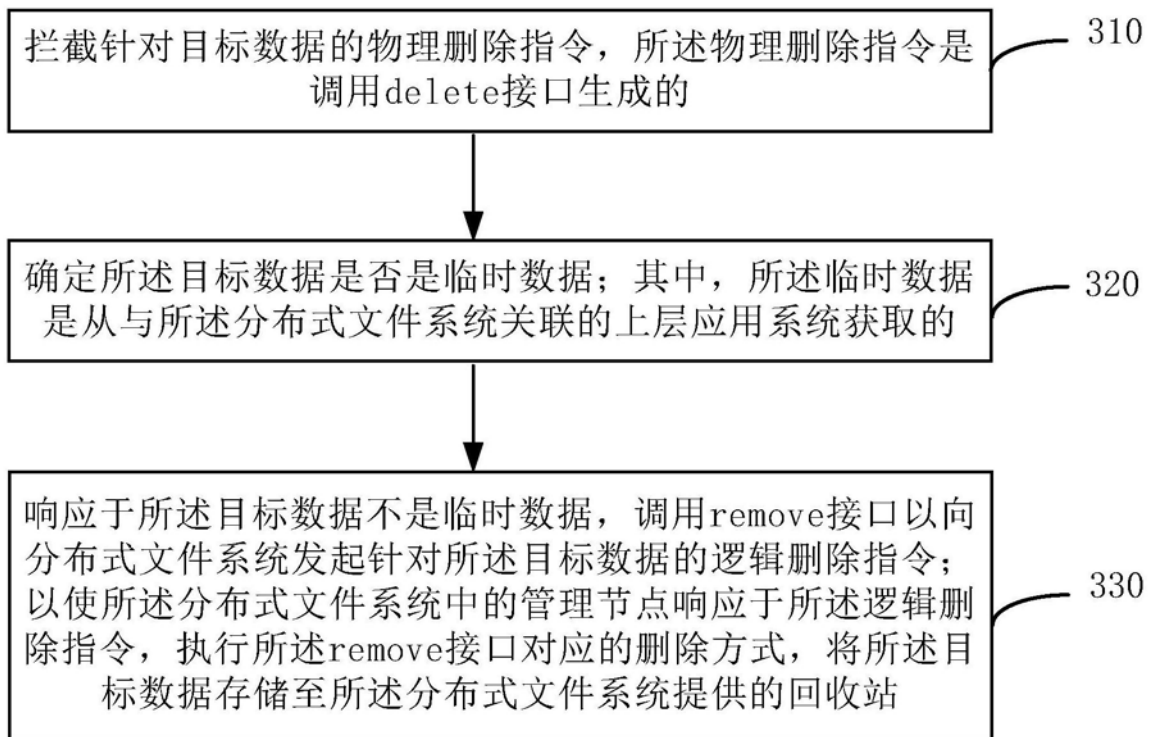


图7

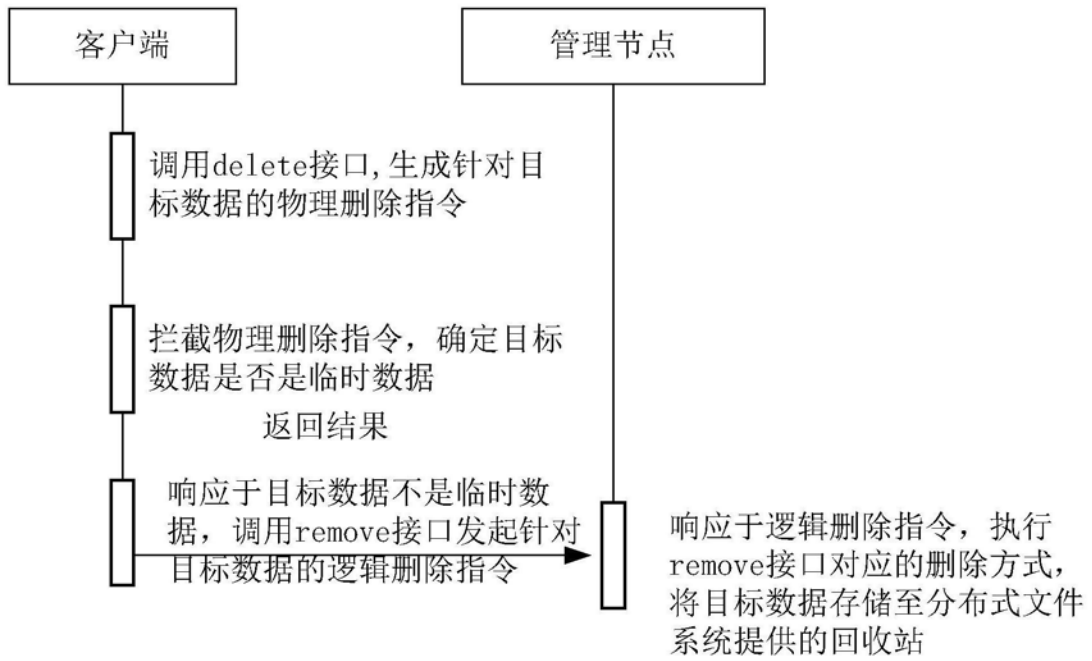


图8



图9

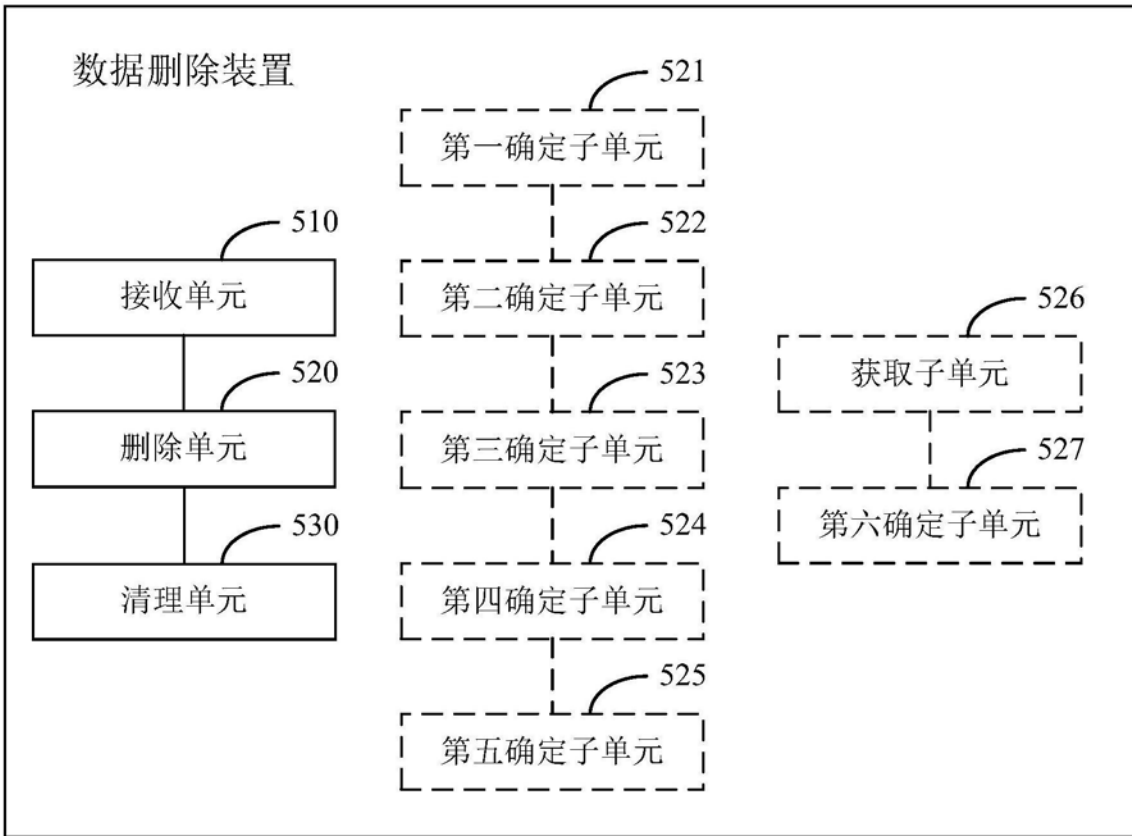


图10

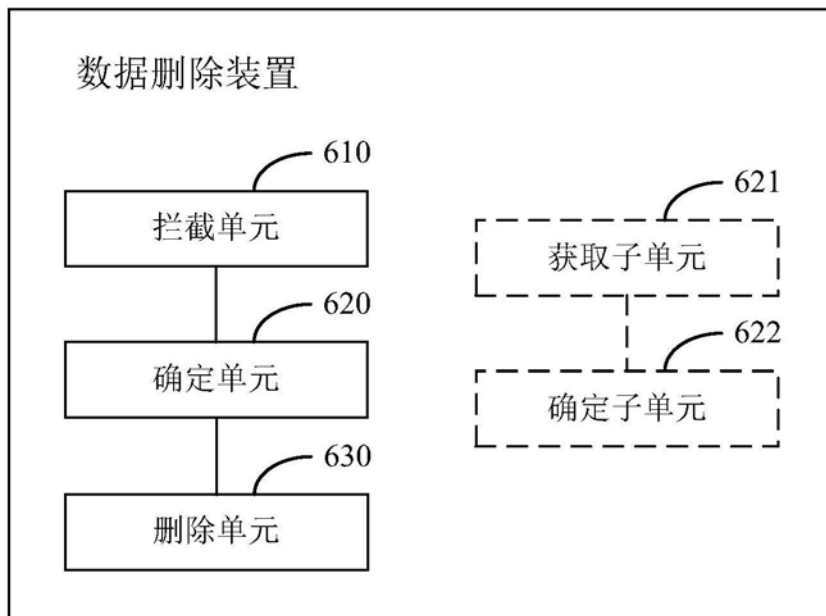


图11

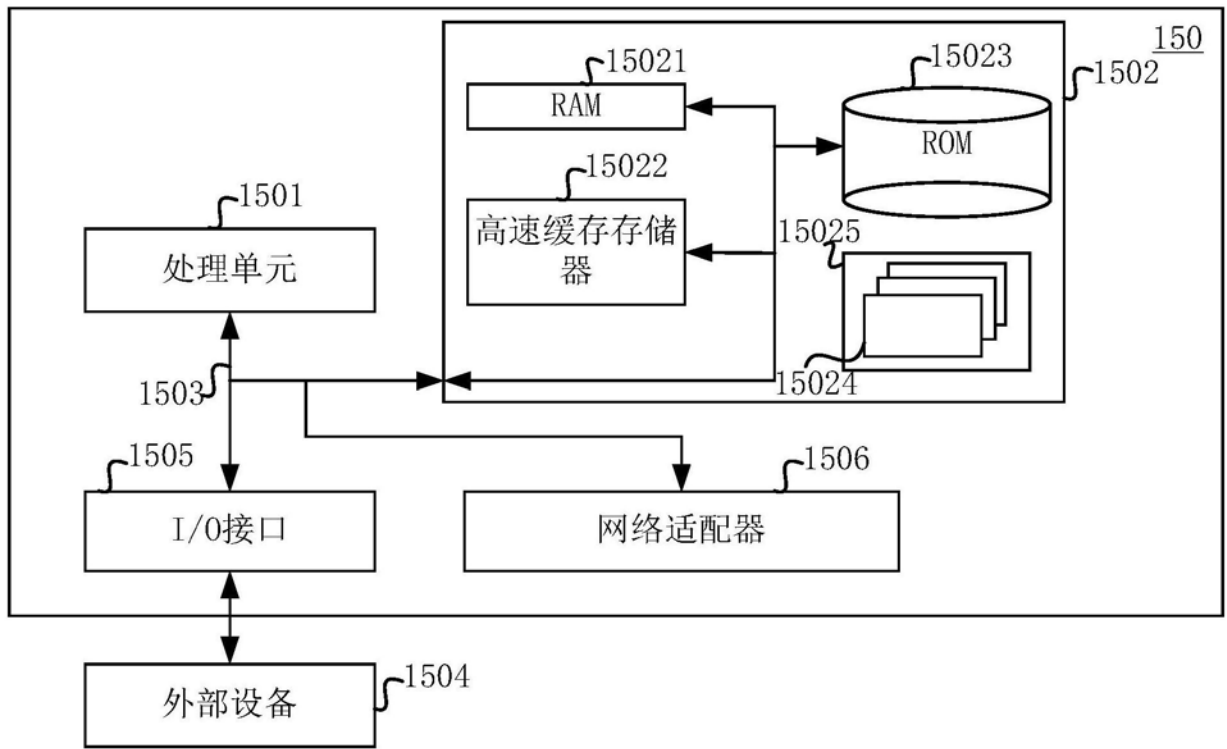


图12