



# (12)发明专利申请

(10)申请公布号 CN 108228770 A

(43)申请公布日 2018.06.29

(21)申请号 201711450749.6

(22)申请日 2017.12.27

(71)申请人 青岛海信移动通信技术股份有限公司

地址 266071 山东省青岛市市南区江西路11号

(72)发明人 孙哲 肖倩

(74)专利代理机构 北京同达信恒知识产权代理有限公司 11291

代理人 黄志华

(51)Int.Cl.

G06F 17/30(2006.01)

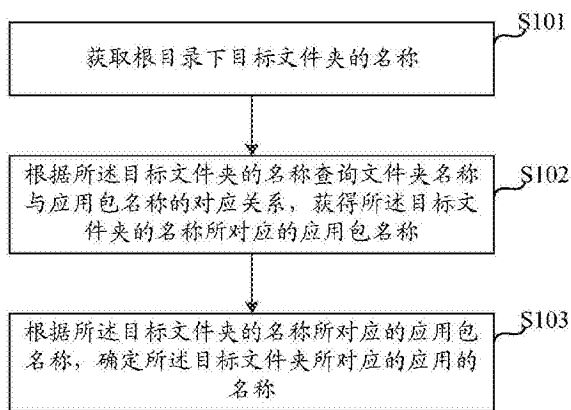
权利要求书2页 说明书8页 附图4页

## (54)发明名称

一种应用文件来源查询的方法及装置

## (57)摘要

本发明公开了一种应用文件来源查询的方法及装置,该方法包括获取根目录下目标文件夹的名称,根据目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得目标文件夹的名称所对应的应用包名称,根据目标文件夹的名称所对应的应用包名称,确定目标文件夹所对应的应用的名称。从而确定出该目标文件夹是由那个应用创建的,查询出该文件的来源,实现了系统应用自动识别出创建文件的真实进程或应用。



1. 一种应用文件来源查询的方法,其特征在于,包括:
  - 获取根目录下目标文件夹的名称;
  - 根据所述目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得所述目标文件夹的名称所对应的应用包名称;
  - 根据所述目标文件夹的名称所对应的应用包名称,确定所述目标文件夹所对应的应用名称。
2. 如权利要求1所述的方法,其特征在于,根据下述步骤确定所述文件夹名称与应用包名称的对应关系,包括:
  - 获取并解析日志文件,确定所述目标文件夹的检测标识所在的行信息;
  - 对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号;
  - 根据所述创建所述目标文件夹的进程号,确定所述进程号对应的应用包名称;
  - 建立所述目标文件夹的名称与所述确定的所述进程号对应的应用包名称的对应关系。
3. 如权利要求2所述的方法,其特征在于,所述对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号,包括:
  - 按照预设第一格式对所述目标文件夹的检测标识所在的行信息进行分解,确定所述目标文件夹的目录路径信息和所述创建所述目标文件夹的进程号;
  - 根据所述目标文件夹的目录路径信息,确定所述目标文件夹的名称。
4. 如权利要求2所述的方法,其特征在于,在所述获取并解析日志文件之前,还包括:
  - 在所述目标文件夹创建时,设定检测标识;
  - 获取所述目标文件夹的目录路径信息和创建所述目标文件夹的进程号;
  - 将所述目标文件夹的检测标识、所述目标文件夹的目录路径信息和所述创建所述目标文件夹的进程号按照第一预设格式进行连接,生成日志文件。
5. 一种应用文件来源查询的装置,其特征在于,包括:
  - 获取单元,用于获取根目录下目标文件夹的名称;
  - 处理单元,用于根据所述目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得所述目标文件夹的名称所对应的应用包名称;以及根据所述目标文件夹的名称所对应的应用包名称,确定所述目标文件夹所对应的应用名称。
6. 如权利要求5所述的装置,其特征在于,所述处理单元在根据下述步骤确定所述文件夹名称与应用包名称的对应关系时,具体用于:
  - 获取并解析日志文件,确定所述目标文件夹的检测标识所在的行信息;
  - 对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号;
  - 根据所述创建所述目标文件夹的进程号,确定所述进程号对应的应用包名称;
  - 建立所述目标文件夹的名称与所述确定的所述进程号对应的应用包名称的对应关系。
7. 如权利要求6所述的装置,其特征在于,所述处理单元在对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号时,具体用于:
  - 按照预设第一格式对所述目标文件夹的检测标识所在的行信息进行分解,确定所述目

标文件夹的目录路径信息和所述创建所述目标文件夹的进程号；

根据所述目标文件夹的目录路径信息，确定所述目标文件夹的名称。

8. 如权利要求6所述的装置，其特征在于，所述处理单元在所述获取并解析日志文件之前，还用于：

在所述目标文件夹创建时，设定检测标识；

获取所述目标文件夹的目录路径信息和创建所述目标文件夹的进程号；

将所述目标文件夹的检测标识、所述目标文件夹的目录路径信息和所述创建所述目标文件夹的进程号按照第一预设格式进行连接，生成日志文件。

9. 一种计算设备，其特征在于，包括：

存储器，用于存储程序指令；

处理器，用于调用所述存储器中存储的程序指令，按照获得的程序执行权利要求1至4任一项所述的方法。

10. 一种计算机存储介质，其特征在于，所述计算机可读存储介质存储有计算机可执行指令，所述计算机可执行指令用于使计算机执行权利要求1至4任一项所述的方法。

## 一种应用文件来源查询的方法及装置

### 技术领域

[0001] 本发明实施例涉及终端技术领域,尤其涉及一种应用文件来源查询的方法及装置。

### 背景技术

[0002] Android(安卓)设备中应用开发自由度非常高,这给我们带来了非常丰富的应用,但也因如此,很多应用开发者并不完全遵从谷歌的要求,例如对于存储空间的访问和使用,应用开发者就非常随意。

[0003] 谷歌原本的设计是希望应用将属于自己的文件创建到终端存储的系统的应用目录下,这样不管是应用还是用户,都可以了解这些文件是谁创建的。

[0004] 然而,现在很多应用都喜欢自己在终端存储的根目录下创建目录而非系统目录,由于命名非常随意,很多目录从名称上完全看不出是哪个应用的,举例如tencent这个目录就是微信、QQ等应用的数据目录,这种还可以大概看出来,但如autonavi这种目录,一般用户根本不知道是谁建立的。

[0005] 从技术上来说,谷歌现有方案中采用了一层虚拟文件系统进行统一的接口权限管理,所有应用调用文件创建的接口后,最终都会走到该文件系统里进行统一管理与管理。结果就是,不管是A应用还是B应用,只要是在终端存储上创建文件或文件夹,最终生成这个文件或文件夹的既不是A也不是B,而是用户root,这一来,想通过查看文件所属的用户组的方式来判断文件来源也是不可能了。

[0006] 目前,很多厂家为了提高用户体验,还是想方设法的多识别一些应用自建的目录,如:autonavi是高德地图应用建立的,baidu是百度搜索建立的,budejie是百思不得姐建立的等等,谁家的统计数据更多,就有更好的兼容性,厂家可以将这个统计数据做成查询列表,让文件管理器之类的应用去查询,并在文件夹名称后面显示个备注,提示用户,例如,tencent|微信文件夹。

[0007] 但这种方式一是更新有延迟,需要服务器端持续更新总表,二是永远无法兼容所有第三方应用。因此,需要一种新的实现方法,让所有应用建立的文件来源都能有法可查,且实时性较好。

### 发明内容

[0008] 本发明实施例提供一种应用文件来源查询的方法及装置,用以实现系统应用自动识别出创建文件的真实进程或应用。

[0009] 本发明实施例提供的一种应用文件来源查询的方法,包括:

[0010] 获取根目录下目标文件夹的名称;

[0011] 根据所述目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得所述目标文件夹的名称所对应的应用包名称;

[0012] 根据所述目标文件夹的名称所对应的应用包名称,确定所述目标文件夹所对应的

应用的名称。

[0013] 可选的,根据下述步骤确定所述文件夹名称与应用包名称的对应关系,包括:

[0014] 获取并解析日志文件,确定所述目标文件夹的检测标识所在的行信息;

[0015] 对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号;

[0016] 根据所述创建所述目标文件夹的进程号,确定所述进程号对应的应用包名称;

[0017] 建立所述目标文件夹的名称与所述确定的所述进程号对应的应用包名称的对应关系。

[0018] 可选的,所述对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号,包括:

[0019] 按照预设第一格式对所述目标文件夹的检测标识所在的行信息进行分解,确定所述目标文件夹的目录路径信息和所述创建所述目标文件夹的进程号;

[0020] 根据所述目标文件夹的目录路径信息,确定所述目标文件夹的名称。

[0021] 可选的,在所述获取并解析日志文件之前,还包括:

[0022] 在所述目标文件夹创建时,设定检测标识;

[0023] 获取所述目标文件夹的目录路径信息和创建所述目标文件夹的进程号;

[0024] 将所述目标文件夹的检测标识、所述目标文件夹的目录路径信息和所述创建所述文件夹的进程号按照第一预设格式进行连接,生成日志文件。

[0025] 相应的,本发明实施例还提供了一种应用文件来源查询的装置,包括:

[0026] 获取单元,用于获取根目录下目标文件夹的名称;

[0027] 处理单元,用于根据所述目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得所述目标文件夹的名称所对应的应用包名称;以及根据所述目标文件夹的名称所对应的应用包名称,确定所述目标文件夹所对应的应用的名称。

[0028] 可选的,所述处理单元在根据下述步骤确定所述文件夹名称与应用包名称的对应关系时,具体用于:

[0029] 获取并解析日志文件,确定所述目标文件夹的检测标识所在的行信息;

[0030] 对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号;

[0031] 根据所述创建所述目标文件夹的进程号,确定所述进程号对应的应用包名称;

[0032] 建立所述目标文件夹的名称与所述确定的所述进程号对应的应用包名称的对应关系。

[0033] 可选的,所述处理单元在对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号时,具体用于:

[0034] 按照预设第一格式对所述目标文件夹的检测标识所在的行信息进行分解,确定所述目标文件夹的目录路径信息和所述创建所述目标文件夹的进程号;

[0035] 根据所述目标文件夹的目录路径信息,确定所述目标文件夹的名称。

[0036] 可选的,所述处理单元在所述获取并解析日志文件之前,还用于:

[0037] 在所述目标文件夹创建时,设定检测标识;

[0038] 获取所述目标文件夹的目录路径信息和创建所述目标文件夹的进程号;

[0039] 将所述目标文件夹的检测标识、所述目标文件夹的目录路径信息和所述创建所述目标文件夹的进程号按照第一预设格式进行连接,生成日志文件。

[0040] 本发明实施例还提供了一种计算设备,包括:

[0041] 存储器,用于存储程序指令;

[0042] 处理器,用于调用所述存储器中存储的程序指令,按照获得的程序执行上述应用文件来源查询的方法。

[0043] 本发明实施例还提供了一种计算机存储介质,所述计算机可读存储介质存储有计算机可执行指令,所述计算机可执行指令用于使计算机执行上述应用文件来源查询的方法。

[0044] 本发明实施例表明,获取根目录下目标文件夹的名称,根据目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得目标文件夹的名称所对应的应用包名称,根据目标文件夹的名称所对应的应用包名称,确定目标文件夹所对应的应用的名称。从而确定出该目标文件夹是由那个应用创建的,查询出该文件的来源,实现了系统应用自动识别出创建文件的真实进程或应用。

## 附图说明

[0045] 为了更清楚地说明本发明实施例中的技术方案,下面将对实施例描述中所需要使用的附图作简要介绍,显而易见地,下面描述中的附图仅仅是本发明的一些实施例,对于本领域的普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0046] 图1为本发明实施例提供了一种应用文件来源查询的方法的流程示意图;

[0047] 图2为本发明实施例提供了一种建立对应关系的流程示意图;

[0048] 图3为本发明实施例提供了一种应用目录创建的流程示意图;

[0049] 图4为本发明实施例提供了一种确定文件夹名称和应用包名称对应关系的流程示意图;

[0050] 图5为本发明实施例提供了一种应用文件来源查询的装置的结构示意图。

## 具体实施方式

[0051] 为了使本发明的目的、技术方案和优点更加清楚,下面将结合附图对本发明作进一步地详细描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其它实施例,都属于本发明保护的范围。

[0052] 本发明实施例提供的应用文件来源查询的方法可以由终端设备执行,该终端设备可以是指向用户提供语音和/或数据连通性的设备(device),包括无线终端或有线终端。无线终端可以是具有无线连接功能的手持式设备、或连接到无线调制解调器的其他处理设备,经无线接入网与一个或多个核心网进行通信的移动终端。例如,无线终端可以是移动电话(或称为“蜂窝”电话)和具有移动终端的计算机。又如,无线终端也可以是便携式、袖珍式、手持式、计算机内置的或者车载的移动装置。再如,无线终端可以为移动站(英文为:mobile station)、接入点(英文为:access point)、或用户设备(英文为:user equipment,

简称UE)的一部分。

[0053] 基于上述描述,图1示例性的示出了本发明实施例提供的一种应用文件来源查询的方法,该方法可以由应用文件来源查询的装置执行,该装置可以是终端设备,也可以位于终端设备内。

[0054] 如图1所示,该流程具体包括:

[0055] 步骤S101,获取根目录下目标文件夹的名称。

[0056] 步骤S102,根据所述目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得所述目标文件夹的名称所对应的应用包名称。

[0057] 步骤S103,根据所述目标文件夹的名称所对应的应用包名称,确定所述目标文件夹所对应的应用的名称。

[0058] 在本发明实施例中,为了便于描述,下面将以文件管理器希望查询到autonavi这个目录是哪个应用创建的为例来描述应用文件来源查询的流程。

[0059] 当文件管理器侧扫描根目录发现有未知文件夹autonavi时,该未知文件夹为目标文件夹,从而可以获知根目录下目标文件夹的名称autonavi。在本发明实施例中,目标文件夹可以为未知文件夹,也就是文件管理器无法根据文件夹的名称识别出其对应的应用名称的文件夹。此时可以根据该autonavi查询文件夹名称与应用包名称的对应关系,该对应关系可以位于公共数据库中,例如可以设置key为文件夹名称,value可以为应用包名称。通过查询key为autonavi,可以得到该key对应的value为com.autonavi.minimap,也就是说,该文件夹名称autonavi对应的应用包名称为com.autonavi.minimap。

[0060] 在得到应用包名称后,文件管理器就可以通过系统接口,获取该应用包名称对应的应用名称“高德地图”,并可以显示出来,便于用户查看。

[0061] 需要说明的是,若获取的value值查询不到应用,说明是个底层服务类进程,可以根据需要作为进程名显示。

[0062] 可选的,为了便于文件管理器通过目标文件夹的名称查询到应用包名称,可以根据如图2所示的流程来确定上述文件夹名称与应用包名称的对应关系。

[0063] 如图2所示,该流程具体步骤包括:

[0064] 步骤S201,获取并解析日志文件,确定目标文件夹的检测标识所在的行信息。

[0065] 由于不同大小的log缓冲区,可以存放的log时间长短不同,例如,可以保存10s,为了能够获取到所有log,可以以小于该存放log时间的任意时间来获取log,例如,可以按照5s一次来获取log。

[0066] 在终端设备的系统启动后,可以启动fileDetectService(文件发现服务)循环的获取log。该服务的作用是循环执行并获取最新的log输出文件。

[0067] 在循环获取log后,对获取的log进行逐行解析,当解析到某一行的内容包含有目标文件夹的检测标识时,说明该行信息中包含有目标文件夹的部分信息。需要对此行进行存储并进行下一步处理。

[0068] 可选的,在获取该预设周期内的log文件之前,还需要先设定目标文件夹的检测标识,也就是说,在该目标文件夹创建时,设定检测标识。然后获取目标文件夹的目录路径信息和创建文件夹的进程号。最后将目标文件夹的检测标识、目标文件夹的目录路径信息和创建该目标文件夹的进程号按照第一预设格式进行连接,生成日志文件。该检测标识用于

对日志文件进行解析时定位到目标文件夹的目录路径信息和创建所述文件夹的进程号所在的行。该检测标识可以是一个固定字符,或者是预设的一段语句等,对此不做限定。该第一预设格式可以是使用标点符号作为语句间隔的格式,或者是其它用于间隔语句的格式,对此不做限制。

[0069] 举例来说,如图3所示的应用创建目录的流程,该流程具体包括:

[0070] 步骤S301,调用file类的mkdir接口。

[0071] 应用高德地图调用File(文件)公共类的mkdir接口创建文件夹,该mkdir接口是一个建立新的文件夹的接口。由于File类是一个基础类,且是会被频繁调用的类,大部分复杂操作都不适合在此添加,如数据库操作等需要上下文的方法也不能使用。在mkdir接口中添加一句log输出。由于mkdir接口的运行空间仍是在应用内部,也就是高德地图内,所以该log输出时所在的进程仍然是应用进程,因此:使用固定字符传作为检测标识。如:“twfx The app is Attempting to make a dir”。

[0072] 步骤S302,获取pid号和获取目标文件路径。

[0073] 通过mkdir接口获取即将创建的目录文件路径。如“/storage/emulated/0/autonavi”。这个完整目录文件路径中还包括了目标文件夹的名称autonavi。然后使用getpid接口获取当前log所在的进程的pid,即进程号。如“5968”,也就是创建该文件夹时的进程号,便于后续查询是由哪个应用创建的文件夹。

[0074] 步骤S303,将检测标识、路径、pid组合为字符串,并将该字符串作为log输出内容。

[0075] 将检测标识、mkdir接口要创建的目录完整路径、进程号三部分内容使用“;”连接起来,作为log输出的内容。最终log输出的内容为:检测标识+路径+pid。举例:twfx The app is Attempting to make a dir;/storage/emulated/0/iFlyIME;5968。

[0076] 步骤S304,创建目录或文件。

[0077] 在输出log文件后,高德地图创建目录或文件,从而目录或文件创建完成。

[0078] 需要说明的是,本发明实施例所举的例子仅是示例作用,对于具体的应用不做限制,终端设备中的所有应用都适用于本发明实施例所提供的方法。

[0079] 步骤S202,对目标文件夹的检测标识所在的行信息进行处理,得到目标文件夹的名称和创建文件夹的进程号。

[0080] 在对目标文件夹的检测标识所在的行信息进行处理,进行处理时,具体可以为首先按照预设第一格式对目标文件夹的检测标识所在的行信息进行分解,得到目标文件夹的目录路径信息和创建所述文件夹的进程号。然后根据目标文件夹的目录路径信息,得到目标文件夹的名称。

[0081] 例如,将包含检测标识的行存储为字符串,进行字符处理,首先将该字符串按照“;”进行分段,按照预设第一格式,最终会分为三段。第一段为检测标识,无需进一步处理。第二段为目标文件夹的目录路径信息,可以得到目标文件夹的完整路径。第三段为创建该文件夹是的进程号。根据第二段的目标文件夹的完整路径就可以得到该目标文件夹的名称。

[0082] 步骤S203,根据创建所述文件夹的进程号,确定进程号对应的应用包名称。

[0083] 在得到创建所述文件夹的进程号后,可以通过pid查询进程所属的应用,并获取应用包名,从而可以获得进程号对应的应用包名称。



[0084] 步骤S204,建立创建的文件夹的名称与确定的进程号对应的应用包名称的对应关系。

[0085] 在得到创建的文件夹的名称与确定的进程号对应的应用包名称之后,就可以建立创建的文件夹的名称与确定的进程号对应的应用包名称的对应关系,并将该对应关系进行存储,例如存储在公共数据库中,便于文件管理器进行查询。

[0086] 举例来说,如图4所示的确定创建的文件夹的名称与确定的进程号对应的应用包名称的对应关系的流程,该流程具体包括:

[0087] 步骤S401,启动检测服务。

[0088] 首先,系统开机后,启动一个服务,举例名为fileDetectService,该服务的作用是循环执行并获取最新的log输出。

[0089] 步骤S402,获取log信息。

[0090] 针对不同大小的log缓冲区,可以存放的log时间长短不同,举例某厂家的缓冲区可以保存10s的log,为了能够获取所有log,则fileDetectService的循环周期可以定位5s。

[0091] 步骤S403,逐行解析log。

[0092] 每次循环,fileDetectService先将最近10s的log获取出来,并逐行解析。

[0093] 步骤S404,判断log中是否存在检测标识,若是,则进入步骤S405,若否,则进入步骤S404。

[0094] 当解析到某一行内容包含检测标识(如“twfx The app is Attempting to make a dir”)时,将此行单独存储并进行下一步处理。如下输出log中,含有检测标识的为第三行:

[0095] 第一行:D twfx-block:startActivityLocked app:com.hmct.vision userid=0unlocked=false should\_lock=false str=null

[0096] 第二行:D twfx-block:isFrontstack stack=Activity Stack{aea47bb stacked=0,1tasks}

[0097] 第三行:E system:twfx the app is Attempting to make a dir;/storage/emulated/0/autonavi;5968

[0098] 该第三行语句也就是检测标识所在的行信息。

[0099] 步骤S405,按标记分割字符串,获取到pid和path。

[0100] 将包含检测标识的行存储为字符串,进行字符处理,首先将该字符串按照“;”进行分段,按照我们的格式,最终会分为三段。第一段为无用内容与检测标识的结合,这部分无需进一步处理。第二段为创建文件的完整路径,记为path。该path为/storage/emulated/0/autonavi。第三段为创建该文件的进程号,记为pid。该pid为5968。

[0101] 步骤S406,获取敏感路径地址。

[0102] fileDetectService服务获取敏感路径地址,此处可以为终端设备存储根目录:“/storage/emulated/0”。

[0103] 步骤S407,判断path中是否包括敏感路径,若是,则转入步骤S408,若否,则转入步骤S404。

[0104] 步骤S408,从path中分离出文件夹的名称。

[0105] 检查path是否包含敏感路径,若包含,则说明该文件夹创建的地方是我们需要进

行查询的位置。将path中的敏感路径去掉,就是文件的相对目录与名称。记为name。

[0106] 步骤S409,获取pid对应的应用包名或进程名。

[0107] 根据pid查询进程所属的应用,并获取包名,此处分两方面:

[0108] 第一方面:使用PackageManager应用程序管理相关接口,获取包名。此种方法可以获取在android framework侧注册过的应用的包名。

[0109] 第二方面:若第一方面无法获取到包名,则可以通过执行PS(Process Status,进程状态)命令,获取应用进程信息,如下:

[0110] u0\_a111 5968 612 1197452 134924 sys\_epoll\_00f3ffc4c8 s  
com.autonavi.minimap。

[0111] PS信息中可以按空格进行字符串分段解析,最后一段就是进程名,假如建立文件的进程不是一个普通应用,而是一个底层服务或者可执行文件之类的时候,通过进程名可以获知该服务的名称。如此处可以获知该应用包名为com.autonavi.minimap。

[0112] 步骤S410,将文件夹的名称和应用包名作为键值对记录到可访问的数据表中。

[0113] 将包名与name作为一个匹配表项记录到一个可查询的区域中,如公共数据库里:格式举例为:

[0114] value(值):com.autonavi.minimap,key(键):autonavi。该key和value为一个键值对。

[0115] 该数据表的存储方式仅是一种示例,对此不做限制,在实际应用时还可以是其它的存储方式来存储上述文件夹名称与应用包名称的对应关系。

[0116] 上述实施例表明,通过获取根目录下目标文件夹的名称,根据目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得目标文件夹的名称所对应的应用包名称,根据目标文件夹的名称所对应的应用包名称,确定目标文件夹所对应的应用的名称。从而确定出该目标文件夹是由那个应用创建的,查询出该文件的来源,实现了系统应用自动识别出创建文件的真实进程或应用。

[0117] 基于相同的技术构思,图5示例性的示出了本发明实施例提供的一种应用文件来源查询的装置,该装置可以执行应用文件来源查询的流程,该装置可以是终端设备,也可以位于终端设备内。

[0118] 如图5所示,该装置具体包括:

[0119] 获取单元501,用于获取根目录下目标文件夹的名称;

[0120] 处理单元502,用于根据所述目标文件夹的名称查询文件夹名称与应用包名称的对应关系,获得所述目标文件夹的名称所对应的应用包名称;以及根据所述目标文件夹的名称所对应的应用包名称,确定所述目标文件夹所对应的应用的名称。

[0121] 可选的,所述处理单元502在根据下述步骤确定所述文件夹名称与应用包名称的对应关系时,具体用于:

[0122] 获取并解析日志文件,确定所述目标文件夹的检测标识所在的行信息;

[0123] 对所述目标文件夹的检测标识所在的行信息进行处理,得到所述目标文件夹的名称和创建所述目标文件夹的进程号;

[0124] 根据所述创建所述目标文件夹的进程号,确定所述进程号对应的应用包名称;

[0125] 建立所述目标文件夹的名称与所述确定的所述进程号对应的应用包名称的对应

关系。

[0126] 可选的,所述处理单元502在对所述目标文件夹的检测标识所在的行信息进行处理,确定所述目标文件夹的名称和创建所述目标文件夹的进程号时,具体用于:

[0127] 按照预设第一格式对所述目标文件夹的检测标识所在的行信息进行分解,确定所述目标文件夹的目录路径信息和所述创建所述目标文件夹的进程号;

[0128] 根据所述目标文件夹的目录路径信息,确定所述目标文件夹的名称。

[0129] 可选的,所述处理单元502在所述获取并解析日志文件之前,还用于:

[0130] 在所述目标文件夹创建时,设定检测标识;

[0131] 获取所述目标文件夹的目录路径信息和创建所述目标文件夹的进程号;

[0132] 将所述目标文件夹的检测标识、所述目标文件夹的目录路径信息和所述创建所述目标文件夹的进程号按照第一预设格式进行连接,生成日志文件。

[0133] 本发明实施例还提供了一种计算设备,包括:存储器,用于存储程序指令;

[0134] 处理器,用于调用所述存储器中存储的程序指令,按照获得的程序执行上述应用文件来源查询的方法。

[0135] 本发明实施例还提供了一种计算机存储介质,所述计算机可读存储介质存储有计算机可执行指令,所述计算机可执行指令用于使计算机执行上述应用文件来源查询的方法。

[0136] 本发明是参照根据本发明实施例的方法、设备(系统)、和计算机程序产品的流程图和/或方框图来描述的。应理解可由计算机程序指令实现流程图和/或方框图中的每一流程和/或方框、以及流程图和/或方框图中的流程和/或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器,使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的装置。

[0137] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中,使得存储在该计算机可读存储器中的指令产生包括指令装置的制造品,该指令装置实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能。

[0138] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上,使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理,从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和/或方框图一个方框或多个方框中指定的功能的步骤。

[0139] 尽管已描述了本发明的优选实施例,但本领域内的技术人员一旦得知了基本创造性概念,则可对这些实施例作出另外的变更和修改。所以,所附权利要求意欲解释为包括优选实施例以及落入本发明范围的所有变更和修改。

[0140] 显然,本领域的技术人员可以对本发明进行各种改动和变型而不脱离本发明的精神和范围。这样,倘若本发明的这些修改和变型属于本发明权利要求及其等同技术的范围之内,则本发明也意图包含这些改动和变型在内。

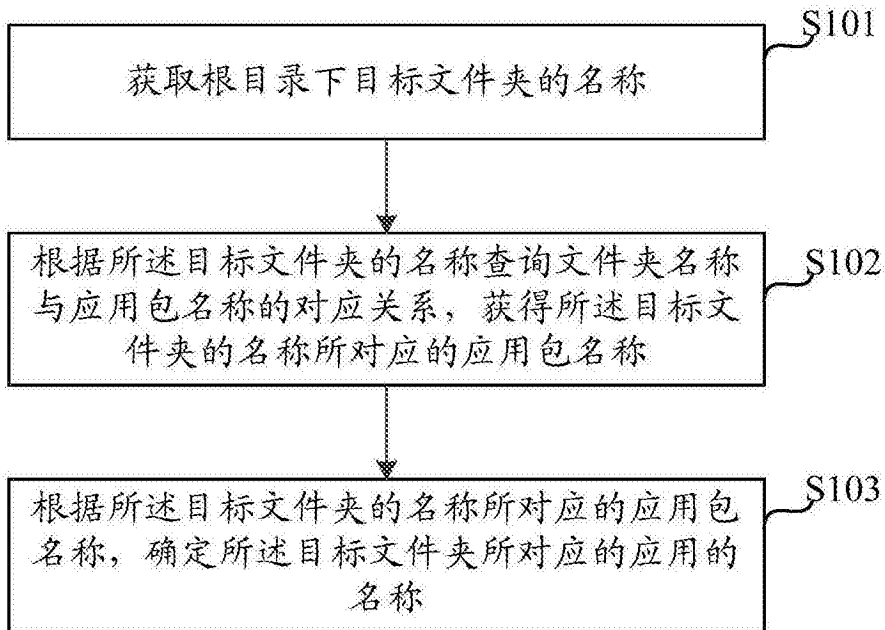


图1

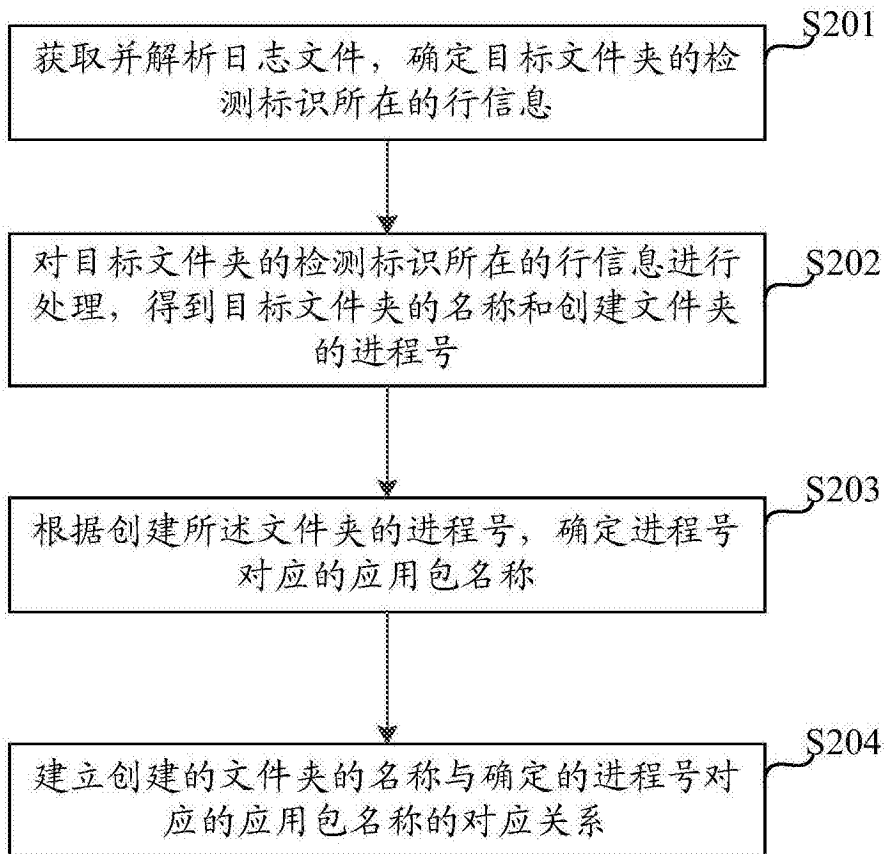


图2

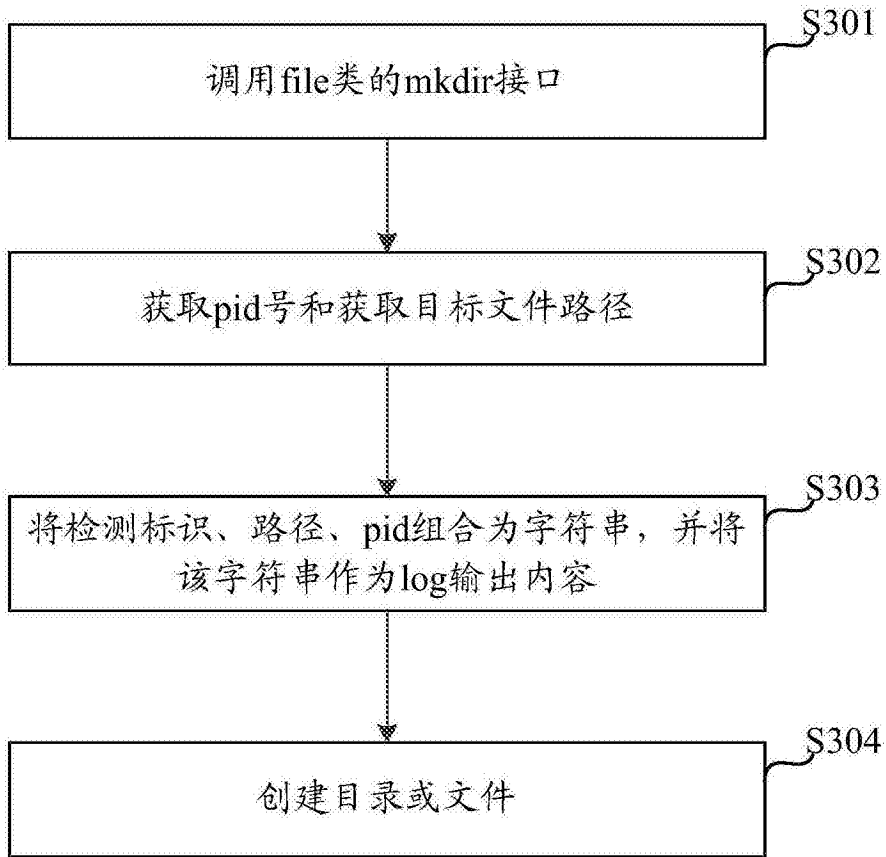


图3

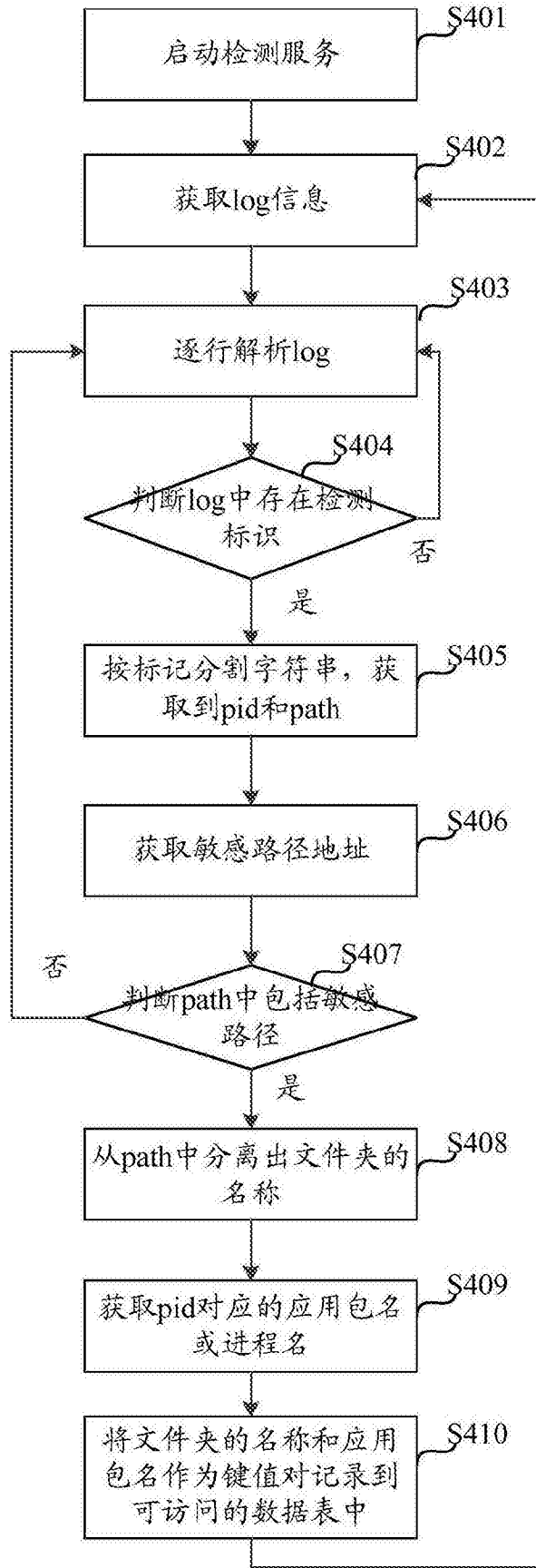


图4

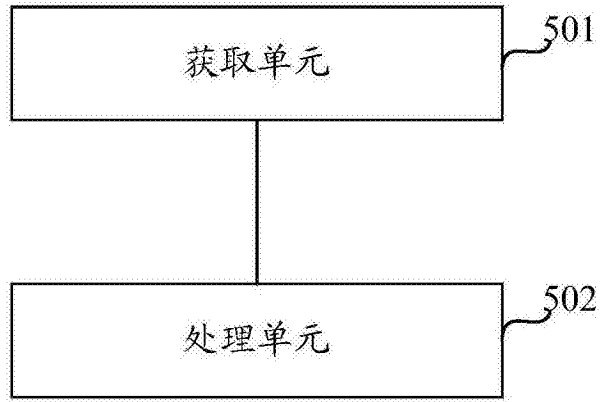


图5