



(12)发明专利申请

(10)申请公布号 CN 110012086 A
(43)申请公布日 2019.07.12

(21)申请号 201910240027.0

(22)申请日 2019.03.27

(71)申请人 努比亚技术有限公司

地址 518000 广东省深圳市南山区高新区
北环大道9018号大族创新大厦A区6-8
层、10-11层、B区6层、C区6-10层

(72)发明人 倪秉炬

(74)专利代理机构 深圳协成知识产权代理事务
所(普通合伙) 44458

代理人 章小燕

(51)Int.Cl.

H04L 29/08(2006.01)

G06F 8/61(2018.01)

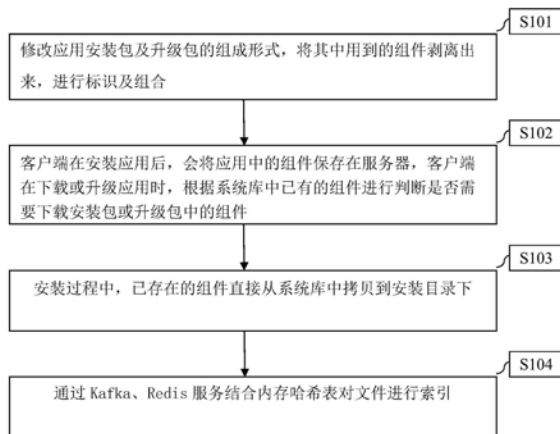
权利要求书1页 说明书11页 附图4页

(54)发明名称

提高应用下载及安装速度的方法、终端及存
储介质

(57)摘要

本发明公开了一种提高应用下载及安装速度的方法,包括:修改应用安装包及升级包的组成形式,将其中用到的组件剥离出来,进行标识及组合、客户端在安装应用后,会将应用中的组件保存在系统库,客户端在下载或升级应用时,根据系统库中已有的组件进行判断是否需要下载安装包或升级包中的某些组件;安装过程中,已存在的组件直接从系统库中拷贝到安装目录下;通过Kafka、Redis服务结合内存哈希表对文件进行索引。这样,一方面可以减少对CDN服务的购买量,较大降低应用中心的系统成本;另一方面,减少了用户在安装或升级应用时的等待时间和消耗流量,提高了用户体验和对手机品牌的认可度。



1. 一种提高应用下载及安装速度的方法,其特征在于,所述方法包括:在服务器中修改应用安装包及升级包的组成形式,将其中的组件剥离出来,并进行标识及组合;客户端在安装应用后,将应用中的组件保存在服务器;客户端在下载或升级应用时,根据服务器中已有的组件进行判断是否需要下载安装包或升级包中的组件;安装过程中,已存在的组件直接从服务器的系统库中拷贝到安装目录下;通过Kafka、Redis服务结合内存哈希表对文件进行索引。

2. 如权利要求1所述的一种提高应用下载及安装速度的方法,其特征在于,在所述通过Kafka、Redis服务结合内存哈希表对文件进行索引这一步骤之前,进一步包括:将所有下载文件存入Kafka服务中,并将其地址作为所有的索引标识。

3. 如权利要求2所述的一种提高应用下载及安装速度的方法,其特征在于,所述客户端在安装应用后,会将应用中的组件保存在服务器,客户端在下载或升级应用时,根据系统库中已有的组件进行判断是否需要下载安装包或升级包中的某些组件,进一步包括:手机客户端从应用中心下载完应用后,会将各个组件保存到单独建立的一个系统组件库中,然后在下载其它应用时,会首先根据自己已经存在的组件及其版本号来判断需要下载应用安装包的何种组件。

4. 如权利要求3所述的一种提高应用下载及安装速度的方法,其特征在于,客户端首先解析该安装包的标识头部分,根据标识头中的组件包名,与自己系统组件库中的包名进行比较,找到重复的部分跳过不下载,只下载自己系统不存在的组件以及安装包的专有内容。

5. 如权利要求4所述的一种提高应用下载及安装速度的方法,其特征在于,客户端通过下载地址解析到组件标识及专有内容地址后,首先根据地址作为key,在应用中心服务的大内存哈希表中查询是否存在该地址对应的内容,如果不存在,则再去Redis服务中以地址为key,去Redis服务的哈希表中查询是否存在文件内容;如果仍然不存在,再根据地址去Kafka服务中查找内容。

6. 如权利要求5所述的一种提高应用下载及安装速度的方法,其特征在于,应用中心会根据各个安装包的实时下载情况,去自动调整安装包存放的位置。

7. 如权利要求6所述的一种提高应用下载及安装速度的方法,其特征在于,应用中心后台服务中设定多个预设值,当下载并发量超过第一预设值时,将安装包的内容放在大内存哈希表中;当下载并发量超过第二预设值时,将安装包的内容放在Redis服务的哈希表中;当并发量小于第三预设值时,安装包的内容只需要放在Kafka服务中。

8. 如权利要求7所述的一种提高应用下载及安装速度的方法,其特征在于,所述第一预设值大于所述第二预设值,所述第二预设值大于所述第三预设值。

9. 一种移动终端,其特征在于,所述移动终端包括处理器以及存储器;

所述存储器用于存储可执行程序;

所述处理器用于执行所述可执行程序以实现如权利要求1-8任一项所述的提高应用下载及安装速度的方法。

10. 一种计算机存储介质,其特征在于,所述计算机存储介质上存储有可执行程序,所述可执行程序被执行时以实现如权利要求1-8任一项所述的提高应用下载及安装速度的方法。

提高应用下载及安装速度的方法、终端及存储介质

技术领域

[0001] 本发明涉及计算机技术领域,尤其涉及一种提高应用下载及安装速度的方法、终端及计算机可读存储介质。

背景技术

[0002] 客户端(Client)是指与网络服务器相对应,与服务器互相配合为用户提供本地服务的软件。在终端设备上运行的客户端,包括基本的操作系统软件和各种应用软件,能够保证终端设备的运行,并向用户提供多种网络服务,如网页浏览、邮件收发和即时通信等。由于通信技术的飞速发展,不断会推出新的操作系统和应用软件,客户端也需要不断地进行升级。现有方案中,当客户端需要升级时,通常是采用单个文件包分别下载的方式,从服务器下载更新后的软件。现有的客户端升级方案至少具有如下缺点:随着网络规模的不断扩展,网络中的用户数量越来越大,甚至能够达到亿级,当大量用户同时升级时,往往会造成网络阻塞,升级速度较慢,然而现有方案还未提出有效的解决方案,并且,现有方案仅支持单个文件包的下载,而文件包中通常不但包括更新后的数据还包括与升级无关的已有数据,造成了网络资源的浪费,也影响了升级的速度。

[0003] 在移动互联网时代,为了更好的满足用户各方面的需求,手机应用的升级频率一般都保持在一个较高的值。为了提高用户在升级应用时的速度,应用新版本发布时,手机厂商一般都会购买较多的CDN服务。

[0004] 现有技术的解决方案存在的以下问题:

[0005] 1. 方案的成本很高。

[0006] 2. 当同一时间升级应用的用户很多时,应用的升级速度会受到影响变慢。

[0007] 3. 用户完整下载安装包的时候,由于安装包往往较大,所以下载时间一般都较长。

[0008] 4. 由于需要安装的组件较多,应用安装或升级的时间都较长。

[0009] 所以,需要实现一套新的方案,以解决这些问题。

发明内容

[0010] 针对上述技术问题,设计了一套提高应用下载及安装速度的方案。该方案的创新点是:1. 修改应用安装包及升级包的组成形式,将其中用到的组件剥离出来,进行标识及组合。

[0011] 2. 手机客户端在安装应用后,会将应用中的组件保存在服务器系统库中。

[0012] 3. 客户端在下载或升级应用时,根据系统库中已有的组件进行判断是否需要下载安装包或升级包中的某些组件,以提高下载速度。

[0013] 4. 安装过程中,已经存在的组件直接从系统库中拷贝到安装目录下,从而提高安装速度。

[0014] 5. 由于组件文件数量非常巨大,为了提高下载速度,引入Kafka、Redis服务,并结合内存哈希表对文件进行索引。

[0015] 将所有下载文件存入Kafka服务中,并将其地址作为所有的索引标识,实现标识一体化。

[0016] 本发明旨在至少解决现有技术中存在的技术问题。为此,本发明公开了一种提高应用下载及安装速度的方法,所述方法包括:

[0017] 步骤S101、修改应用安装包及升级包的组成形式,将其中用到的组件剥离出来,进行标识及组合;

[0018] 步骤S102、客户端在安装应用后,会将应用中的组件保存在系统库,客户端在下载或升级应用时,根据系统库中已有的组件进行判断是否需要下载安装包或升级包中的某些组件;

[0019] 步骤S103、安装过程中,已存在的组件直接从系统库中拷贝到安装目录下;

[0020] 步骤S104、通过Kafka、Redis服务结合内存哈希表对文件进行索引。

[0021] 可选地,将所有下载文件存入Kafka服务中,并将其地址作为所有的索引标识。

[0022] 可选地,所述步骤S102进一步包括:手机客户端从应用中心下载完应用后,会将各个组件保存到单独建立的一个系统组件库中,然后在下载其它应用时,会首先根据自己已经存在的组件及其版本号来判断需要下载应用安装包的何种组件。

[0023] 可选地,所述步骤S102还包括:客户端首先解析该安装包的标识头部分,根据标识头中的组件包名,与自己系统组件库中的包名进行比较,找到重复的部分跳过不下载,只下载自己系统不存在的组件以及安装包的专有内容。

[0024] 可选地,客户端通过下载地址解析到组件标识及专有内容地址后,首先根据地址作为key,在应用中心服务的大内存哈希表中查询是否存在该地址对应的内容,如果不存在,则再去Redis服务中以地址为key,去Redis服务的哈希表中查询是否存在文件内容;如果仍然不存在,再根据地址去Kafka服务中查找内容。

[0025] 可选地,应用中心会根据各个安装包的实时下载情况,去自动调整安装包存放的位置。

[0026] 可选地,应用中心后台服务中设定多个预设值,当下载并发量超过第一预设值时,将安装包的内容放在大内存哈希表中;当下载并发量超过第二预设值时,将安装包的内容放在Redis服务的哈希表中;当并发量小于第三预设值时,安装包的内容只需要放在Kafka服务中。

[0027] 可选地,所述第一预设值大于所述第二预设值,所述第二预设值大于所述第三预设值。

[0028] 本发明还公开了一种电子装置(移动终端),包括:处理器;以及,存储器,用于存储所述处理器的可执行指令;其中,所述处理器配置为经由执行所述可执行指令来执行上述的提高应用下载及安装速度的方法。

[0029] 本发明还公开了一种计算机可读存储介质,其上存储有计算机程序,所述计算机程序被处理器执行时实现上述的提高应用下载及安装速度的方法。

[0030] 本发明的有益效果:通过上述方案,通过改变应用安装或升级包的组成形式,并引入系统组件库,改变客户端的下载机制,同时,引入Kafka服务、Redis服务和大内存哈希表,通过地址一体化索引机制,自动调整文件内容位置,提高文件下载的并发量,从而大大提高了应用的下载和安装速度。这样,一方面可以减少对CDN服务的购买量,较大降低应用中心

的系统成本；另一方面，减少了用户在安装或升级应用时的等待时间和消耗流量，提高了用户体验和对手机品牌的认可度。

附图说明

[0031] 从以下结合附图的描述可以进一步理解本发明。图中的部件不一定按比例绘制，而是将重点放在示出实施例的原理上。在图中，在不同的视图中，相同的附图标记指定对应的部分。

[0032] 图1为实现本发明各个实施例的移动终端的硬件结构示意图；

[0033] 图2为本发明实施例提供的一种通信网络系统架构图；

[0034] 图3是本发明的一种实施例中的提高应用下载及安装速度的方法的流程图；

[0035] 图4是本发明的一种实施例中所示的服务器的结构图；

[0036] 图5是本发明的一种实施例中所示的客户端的结构图。

具体实施方式

[0037] 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份实施例，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。

[0038] 下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例是本发明一部分实施例，而不是全部的实施例。需要说明的是，结合附图所阐述的详细描述旨在作为对各种配置的描述，而不旨在表示其中可以实践本文所描述的概念的唯一配置。本文中所记载的装置实施例和方法实施例将在下面的详细描述中进行描述，并在附图中通过各种框、模块、单元、组件、电路、步骤、过程、算法等等（统称为“要素”）来予以示出。这些要素可以使用电子硬件、计算机软件或者其任意组合来实现。至于这些要素是实现为硬件还是软件，取决于特定应用和施加在整体系统上的设计约束。本发明的说明书和权利要求书以及说明书附图中的术语如果使用“第一”、“第二”等描述，该种描述是用于区别不同对象，而不是用于描述特定顺序。

[0039] 应当理解，当在本说明书和所附权利要求书中使用时，术语“包括”和“包含”指示所描述特征、整体、步骤、操作、元素和/或组件的存在，但并不排除一个或多个其它特征、整体、步骤、操作、元素、组件和/或其集合的存在或添加。还应当理解，在此本发明说明书中所使用的术语仅仅是出于描述特定实施例的目的而并不意在限制本发明。如在本发明说明书和所附权利要求书中所使用的那样，除非上下文清楚地指明其它情况，否则单数形式的“一”、“一个”及“该”意在包括复数形式。还应当进一步理解，在本发明说明书和所附权利要求书中使用的术语“和/或”是指相关联列出的项中的一个或多个的任何组合以及所有可能组合，并且包括这些组合。

[0040] 应当理解，此处所描述的具体实施例仅仅用以解释本发明，并不用于限定本发明。

[0041] 在后续的描述中，使用用于表示元件的诸如“模块”、“部件”或“单元”的后缀仅为为了有利于本发明的说明，其本身没有特定的意义。因此，“模块”、“部件”或“单元”可以混合地使用。

[0042] 终端可以以各种形式来实施。例如,本发明中描述的终端可以包括诸如手机、平板电脑、笔记本电脑、掌上电脑、个人数字助理(Personal Digital Assistant,PDA)、便捷式媒体播放器(Portable Media Player,PMP)、导航装置、可穿戴设备、智能手环、计步器等移动终端,以及诸如数字TV、台式计算机等固定终端。

[0043] 后续描述中将以移动终端为例进行说明,本领域技术人员将理解的是,除了特别用于移动目的的元素之外,根据本发明的实施方式的构造也能够应用于固定类型的终端。

[0044] 请参阅图1,其为实现本发明各个实施例的一种移动终端的硬件结构示意图,该移动终端100可以包括:RF(Radio Frequency,射频)单元101、WiFi模块102、音频输出单元103、A/V(音频/视频)输入单元104、传感器105、显示单元106、用户输入单元107、接口单元108、存储器109、处理器110、以及电源111等部件。本领域技术人员可以理解,图1中示出的移动终端结构并不构成对移动终端的限定,移动终端可以包括比图示更多或更少的部件,或者组合某些部件,或者不同的部件布置。

[0045] 下面结合图1对移动终端的各个部件进行具体的介绍:

[0046] 射频单元101可用于收发信息或通话过程中,信号的接收和发送,具体的,将基站的下行信息接收后,给处理器110处理;另外,将上行的数据发送给基站。通常,射频单元101包括但不限于天线、至少一个放大器、收发信机、耦合器、低噪声放大器、双工器等。此外,射频单元101还可以通过无线通信与网络和其他设备通信。上述无线通信可以使用任一通信标准或协议,包括但不限于GSM(Global System of Mobile communication,全球移动通讯系统)、GPRS(General Packet Radio Service,通用分组无线服务)、CDMA2000(Code Division Multiple Access 2000,码分多址2000)、WCDMA(Wideband Code Division Multiple Access,宽带码分多址)、TD-SCDMA(Time Division-Synchronous Code Division Multiple Access,时分同步码分多址)、FDD-LTE(Frequency Division Duplexing-Long Term Evolution,频分双工长期演进)和TDD-LTE(Time Division Duplexing-Long Term Evolution,分时双工长期演进)等。

[0047] WiFi属于短距离无线传输技术,移动终端通过WiFi模块102可以帮助用户收发电子邮件、浏览网页和访问流式媒体等,它为用户提供了无线的宽带互联网访问。虽然图1示出了WiFi模块102,但是可以理解的是,其并不属于移动终端的必须构成,完全可以根据需要在不改变发明的本质的范围内而省略。

[0048] 音频输出单元103可以在移动终端100处于呼叫信号接收模式、通话模式、记录模式、语音识别模式、广播接收模式等等模式下时,将射频单元101或WiFi模块102接收的或者在存储器109中存储的音频数据转换成音频信号并且输出为声音。而且,音频输出单元103还可以提供与移动终端100执行的特定功能相关的音频输出(例如,呼叫信号接收声音、消息接收声音等等)。音频输出单元103可以包括扬声器、蜂鸣器等等。

[0049] A/V输入单元104用于接收音频或视频信号。A/V输入单元104可以包括图形处理器(Graphics Processing Unit,GPU)1041和麦克风1042,图形处理器1041对在视频捕获模式或图像捕获模式中由图像捕获装置(如摄像头)获得的静态图片或视频的图像数据进行处理。处理后的图像帧可以显示在显示单元106上。经图形处理器1041处理后的图像帧可以存储在存储器109(或其它存储介质)中或者经由射频单元101或WiFi模块102进行发送。麦克风1042可以在电话通话模式、记录模式、语音识别模式等等运行模式中经由麦克风1042接

收声音(音频数据),并且能够将这样的声音处理为音频数据。处理后的音频(语音)数据可以在电话通话模式的情况下转换为可经由射频单元101发送到移动通信基站的格式输出。麦克风1042可以实施各种类型的噪声消除(或抑制)算法以消除(或抑制)在接收和发送音频信号的过程中产生的噪声或者干扰。

[0050] 移动终端100还包括至少一种传感器105,比如光传感器、运动传感器以及其他传感器。具体地,光传感器包括环境光传感器及接近传感器,其中,环境光传感器可根据环境光线的明暗来调节显示面板1061的亮度,接近传感器可在移动终端100移动到耳边时,关闭显示面板1061和/或背光。作为运动传感器的一种,加速计传感器可检测各个方向上(一般为三轴)加速度的大小,静止时可检测出重力的大小及方向,可用于识别手机姿态的应用(比如横竖屏切换、相关游戏、磁力计姿态校准)、振动识别相关功能(比如计步器、敲击)等;至于手机还可配置的指纹传感器、压力传感器、虹膜传感器、分子传感器、陀螺仪、气压计、湿度计、温度计、红外线传感器等其他传感器,在此不再赘述。

[0051] 显示单元106用于显示由用户输入的信息或提供给用户的信息。显示单元106可包括显示面板1061,可以采用液晶显示器(Liquid Crystal Display,LCD)、有机发光二极管(Organic Light-Emitting Diode,OLED)等形式来配置显示面板1061。

[0052] 用户输入单元107可用于接收输入的数字或字符信息,以及产生与移动终端的用户设置以及功能控制有关的键信号输入。具体地,用户输入单元107可包括触控面板1071以及其他输入设备1072。触控面板1071,也称为触摸屏,可收集用户在其上或附近的触摸操作(比如用户使用手指、触笔等任何适合的物体或附件在触控面板1071上或在触控面板1071附近的操作),并根据预先设定的程式驱动相应的连接装置。触控面板1071可包括触摸检测装置和触摸控制器两个部分。其中,触摸检测装置检测用户的触摸方位,并检测触摸操作带来的信号,将信号传送给触摸控制器;触摸控制器从触摸检测装置上接收触摸信息,并将它转换成触点坐标,再送给处理器110,并能接收处理器110发来的命令并加以执行。此外,可以采用电阻式、电容式、红外线以及表面声波等多种类型实现触控面板1071。除了触控面板1071,用户输入单元107还可以包括其他输入设备1072。具体地,其他输入设备1072可以包括但不限于物理键盘、功能键(比如音量控制按键、开关按键等)、轨迹球、鼠标、操作杆等中的一种或多种,具体此处不做限定。

[0053] 进一步的,触控面板1071可覆盖显示面板1061,当触控面板1071检测到在其上或附近的触摸操作后,传送给处理器110以确定触摸事件的类型,随后处理器110根据触摸事件的类型在显示面板1061上提供相应的视觉输出。虽然在图1中,触控面板1071与显示面板1061是作为两个独立的部件来实现移动终端的输入和输出功能,但是在某些实施例中,可以将触控面板1071与显示面板1061集成而实现移动终端的输入和输出功能,具体此处不做限定。

[0054] 接口单元108用作至少一个外部装置与移动终端100连接可以通过的接口。例如,外部装置可以包括有线或无线头戴式耳机端口、外部电源(或电池充电器)端口、有线或无线数据端口、存储卡端口、用于连接具有识别模块的装置的端口、音频输入/输出(I/O)端口、视频I/O端口、耳机端口等等。接口单元108可以用于接收来自外部装置的输入(例如,数据信息、电力等等)并且将接收到的输入传输到移动终端100内的一个或多个元件或者可以用于在移动终端100和外部装置之间传输数据。

[0055] 存储器109可用于存储软件程序以及各种数据。存储器109可主要包括存储程序区和存储数据区,其中,存储程序区可存储操作系统、至少一个功能所需的应用程序(比如声音播放功能、图像播放功能等)等;存储数据区可存储根据手机的使用所创建的数据(比如音频数据、电话本等)等。此外,存储器109可以包括高速随机存取存储器,还可以包括非易失性存储器,例如至少一个磁盘存储器件、闪存器件、或其他易失性固态存储器件。

[0056] 处理器110是移动终端的控制中心,利用各种接口和线路连接整个移动终端的各个部分,通过运行或执行存储在存储器109内的软件程序和/或模块,以及调用存储在存储器109内的数据,执行移动终端的各种功能和处理数据,从而对移动终端进行整体监控。处理器110可包括一个或多个处理单元;优选的,处理器110可集成应用处理器和调制解调处理器,其中,应用处理器主要处理操作系统、用户界面和应用程序等,调制解调处理器主要处理无线通信。可以理解的是,上述调制解调处理器也可以不集成到处理器110中。

[0057] 移动终端100还可以包括给各个部件供电的电源111(比如电池),优选的,电源111可以通过电源管理系统与处理器110逻辑相连,从而通过电源管理系统实现管理充电、放电、以及功耗管理等功能。

[0058] 尽管图1未示出,移动终端100还可以包括蓝牙模块等,在此不再赘述。

[0059] 为了便于理解本发明实施例,下面对本发明的移动终端所基于的通信网络系统进行描述。

[0060] 请参阅图2,图2为本发明实施例提供的一种通信网络系统架构图,该通信网络系统为通用移动通信技术的LTE系统,该LTE系统包括依次通讯连接的UE(User Equipment,用户设备)201,E-UTRAN(Evolved UMTS Terrestrial Radio Access Network,演进式UMTS陆地无线接入网)202,EPC(Evolved Packet Core,演进式分组核心网)203和运营商的IP业务204。

[0061] 具体地,UE201可以是上述终端100,此处不再赘述。

[0062] E-UTRAN202包括eNodeB2021和其它eNodeB2022等。其中,eNodeB2021可以通过回程(backhaul)(例如X2接口)与其它eNodeB2022连接,eNodeB2021连接到EPC203,eNodeB2021可以提供UE201到EPC203的接入。

[0063] EPC203可以包括MME(Mobility Management Entity,移动性管理实体)2031,HSS(Home Subscriber Server,归属用户服务器)2032,其它MME2033,SGW(Serving Gate Way,服务网关)2034,PGW(PDN Gate Way,分组数据网络网关)2035和PCRF(Policy and Charging Rules Function,政策和资费功能实体)2036等。其中,MME2031是处理UE201和EPC203之间信令的控制节点,提供承载和连接管理。HSS2032用于提供一些寄存器来管理诸如归属位置寄存器(图中未示)之类的功能,并且保存有一些有关服务特征、数据速率等用户专用的信息。所有用户数据都可以通过SGW2034进行发送,PGW2035可以提供UE 201的IP地址分配以及其它功能,PCRF2036是业务数据流和IP承载资源的策略与计费控制策略决策点,它为策略与计费执行功能单元(图中未示)选择及提供可用的策略和计费控制决策。

[0064] IP业务204可以包括因特网、内联网、IMS(IP Multimedia Subsystem,IP多媒体子系统)或其它IP业务等。

[0065] 虽然上述以LTE系统为例进行了介绍,但本领域技术人员应当知晓,本发明不仅仅适用于LTE系统,也可以适用于其他无线通信系统,例如GSM、CDMA2000、WCDMA、TD-SCDMA以

及未来新的网络系统等,此处不做限定。

[0066] 基于上述移动终端硬件结构以及通信网络系统,提出本发明方法各个实施例。

[0067] 实施例1

[0068] 如图3所示,一种提高应用下载及安装速度的方法,所述方法包括:

[0069] 在步骤S101中,修改应用安装包及升级包的组成形式,将其中用到的组件剥离出来,进行标识及组合;

[0070] 在步骤S102中,客户端在安装应用后,会将应用中的组件保存在系统库,客户端在下载或升级应用时,根据系统库中已有的组件进行判断是否需要下载安装包或升级包中的某些组件;

[0071] 在步骤S103中,安装过程中,已存在的组件直接从系统库中拷贝到安装目录下;

[0072] 在步骤S104中,通过Kafka、Redis服务结合内存哈希表对文件进行索引。

[0073] 更进一步地,将所有下载文件存入Kafka服务中,并将其地址作为所有的索引标识。

[0074] 更进一步地,所述步骤S102进一步包括:手机客户端从应用中心下载完应用后,会将各个组件保存到单独建立的一个系统组件库中,然后在下载其它应用时,会首先根据自己已经存在的组件及其版本号来判断需要下载应用安装包的何种组件。

[0075] 更进一步地,所述步骤S102还包括:客户端首先解析该安装包的标识头部分,根据标识头中的组件包名,与自己系统组件库中的包名进行比较,找到重复的部分跳过不下载,只下载自己系统不存在的组件以及安装包的专有内容。

[0076] 更进一步地,客户端通过下载地址解析到组件标识及专有内容地址后,首先根据地址作为key,在应用中心服务的大内存哈希表中查询是否存在该地址对应的内容,如果不存在,则再去Redis服务中以地址为key,去Redis服务的哈希表中查询是否存在文件内容;如果仍然不存在,再根据地址去Kafka服务中查找内容。

[0077] 更进一步地,应用中心会根据各个安装包的实时下载情况,去自动调整安装包存放的位置。

[0078] 更进一步地,应用中心后台服务中设定多个预设值,当下载并发量超过第一预设值时,将安装包的内容放在大内存哈希表中;当下载并发量超过第二预设值时,将安装包的内容放在Redis服务的哈希表中;当并发量小于第三预设值时,安装包的内容只需要放在Kafka服务中。

[0079] 更进一步地,所述第一预设值大于所述第二预设值,所述第二预设值大于所述第三预设值。

[0080] 本发明还公开了一种电子装置,包括:处理器;以及,存储器,用于存储所述处理器的可执行指令;其中,所述处理器配置为经由执行所述可执行指令来执行上述的提高应用下载及安装速度的方法。

[0081] 本发明还公开了一种计算机可读存储介质,其上存储有计算机程序,所述计算机程序被处理器执行时实现上述的提高应用下载及安装速度的方法。

[0082] 目前各个手机厂商一般都有自己的应用中心,应用中心中的应用安装包和升级包中都会包括较多的通用组件,例如某些开源组件,例如某些自有组件,并以jar包的形式存在。这些jar包在应用打包时,统一压缩到安装或升级包中。客户端在下载安装包或升级包

时,需要完整下载,当安装包或升级包较大时,下载时间就会相对较长。为了提高应用的下载速度,对应用中心中应用安装或升级包的组成形式进行修改,即将需要用到的通用或自有组件进行标识,然后依次压缩再组合到安装包或升级包中。例如一个安装包有100个组件,分别是组件A、组件B、组件C到组件XX。然后对这些组件依次进行压缩,并在压缩文件前添加组件包名的标识。组合完后,安装包的组成形式即为组件A标识+组件B标识+组件C标识+...组件XX标识+组件A压缩包+组件B压缩包+组件C压缩包...+组件XX压缩包+专有内容。即标识头加组件压缩包加专有内容。组件标识包括组件包名、组件版本号。

[0083] 安装包各文件的压缩包首先存放到Kafka服务中,Kafka服务作为日志服务,拥有巨大的空间和快速的寻址能力。所以通过Kafka服务来存放安装包的各个文件,可以很快的找到各个文件。这样,安装包下载地址中存放的信息就改进为组件A标识+组件A地址+组件B标识+组件B地址...+专有内容地址。

[0084] 手机客户端从应用中心下载完应用后,会将各个组件保存到单独建立的一个系统组件库中。然后在下载其它应用时,会首先根据自己已经存在的组件及其版本号来判断需要下载该应用安装包中的哪些部分。下载应用时,客户端首先解析该安装包的标识头部分,根据标识头中的组件包名,与自己系统组件库中的包名进行比较,找到重复的部分跳过不下载,只下载自己系统不存在的组件以及安装包的专有内容。由于各个应用都是基于安卓系统,基于很多的开源组件进行开发,所以很可能会存在较多重复的组件。这样,例如下载某个应用,该应用有200个组件包,客户端在下载时解析安卓包的标识头部分,发现有160个组件在自己的系统组件库中都存在,于是只需要下载40个组件包和该安装包的专有内容即可,这样真正需要下载的安装包内容就得到大大的减少,从而极大提高了应用的下载速度。同样的,在安装过程中,这些已经存在的组件直接从系统组件库中拷贝的安装目录下,省去了解压组件包的时间,也可以提高应用的安装速度。

[0085] 为了进一步提高应用的下载速度,客户端通过下载地址解析到组件标识及专有内容地址后,首先根据地址作为key,在应用中心服务的大内存哈希表中查询是否存在该地址对应的内容,如果不存在,则再去Redis服务中以地址为key,去Redis服务的哈希表中查询是否存在文件内容;如果仍然不存在,再根据地址去Kafka服务中查找内容。

[0086] 应用中心会根据各个安装包的实时下载情况,去自动调整安装包存放的位置。在应用中心后台服务中会设定阈值,例如下载并发量超过10000时,就会将安装包的内容放在大内存哈希表中,以得到最大的并发量;例如并发量大于5000小于10000时,就将安装包的内容放在Redis服务的哈希表中,以得到适中的并发量;例如并发量小于5000时,安装包的内容只需要放在Kafka服务中即可,以实现服务的成本最低化,同时,也仍然可以保证足够的并发量和下载速度。

[0087] 通过上述方案,通过改变应用安装或升级包的组成形式,并引入系统组件库,改变客户端的下载机制,同时,引入Kafka服务、Redis服务和大内存哈希表,通过地址一体化索引机制,自动调整文件内容位置,提高文件下载的并发量,从而大大提高了应用的下载和安装速度。这样,一方面可以减少对CDN服务的购买量,较大降低应用中心的系统成本;另一方面,减少了用户在安装或升级应用时的等待时间和消耗流量,提高了用户体验和对手机品牌的认可度。

[0088] 实施例二

[0089] 在本实施例中,主要从硬件的角度描述本发明,提供一种提高应用下载及安装速度的系统,所述包括服务器及客户端,所述服务器包括系统库;

[0090] 如图4所示的服务器包括:

[0091] 组件提取模块10,修改应用安装包及升级包的组成形式,将其中的组件剥离出来,进行标识及组合;

[0092] 组件库模块20,客户端在安装应用后,将应用中的组件保存在服务器的系统库;

[0093] 发送模块30:发送模块执行与客户端的通信,将组件库模块中的组件发送至客户端;

[0094] 索引模块40,通过Kafka、Redis服务结合内存哈希表对文件进行索引。

[0095] 组件索引模块还包括:将所有下载文件存入Kafka服务中,并将其地址作为所有的索引标识。调用过Kafka服务加快索引找到定位组件的速度。

[0096] 如图5所示的客户端包括:

[0097] 在下载或升级应用时,所述客户端通过接受模块60接受服务器组件,判断模块80根据服务器系统库中已有的组件进行判断是否需要下载安装包或升级包中的某些组件;

[0098] 组件安装模块50,安装过程中,将已存在服务器系统库的组件直接从系统库中拷贝到客户端安装目录下;

[0099] 在本地组件库模块70中,手机客户端从应用中心下载完应用后,会将各个组件保存到单独建立的一个系统组件库中,然后在下载其它应用时,会首先根据自己已经存在的组件及其版本号来判断需要下载应用安装包的何种组件,客户端在安装程序的版本号确定组件的版本号选择。

[0100] 在本实施例中,可以设置管理数据库来记录应用所用组件信息,并将关系数据库的数据导入到非关系型数据库与索引服务器中;封装所述关系数据库的所述数据,生成索引请求;以及将所述索引请求按照预定Mapping存储结构索引到所述索引服务器中,,将关系数据库的数据导入到非关系型数据库与索引服务器中,将所述关系数据库的所述数据通过Kafka组件,导入到所述非关系型数据库与所述索引服务器中。

[0101] 优选地,索引服务器通过redis访问非关系型数据库中的数据,将关系数据库类型的数据,处理为对应的索引类型的数据。

[0102] 优选地,当所述索引的分片中文档数量大于一预定数值时,添加具有相同别名的索引。

[0103] 本实施例所提供的系统进一步地,客户端首先解析该安装包的标识头部分,根据标识头中的组件包名,与自己系统组件库中的包名进行比较,找到重复的部分跳过不下载,只下载自己系统不存在的组件以及安装包的专有内容。

[0104] 本实施例中,还可以对具有相同功能的可替换的组件进行归集,在自己的系统组件库中如果具有与需要下载的安装包中相类似的组件,则用户可以选择是否下载此组件,如果用户选择下载,则按照常规组件进行下载,如果用户不选择下载该组件,则在本地的系统组件库中选择功能相近可以替换的组件进行应用的安装。

[0105] 本实施例中,为了提高应用的下载和安装的速度,客户端通过下载地址解析到组件标识及专有内容地址后,首先根据地址作为key,在应用中心服务的大内存哈希表中查询是否存在该地址对应的内容,如果不存在,则再去Redis服务中以地址为key,去Redis服务

的哈希表中查询是否存在文件内容;如果仍然不存在,再根据地址去Kafka服务中查找内容。

[0106] 应用中心会根据各个安装包的实时下载情况,去自动调整安装包存放的位置。

[0107] 应用中心后台服务中设定多个预设值,当下载并发量超过第一预设值时,将安装包的内容放在大内存哈希表中;当下载并发量超过第二预设值时,将安装包的内容放在Redis服务的哈希表中;当并发量小于第三预设值时,安装包的内容只需要放在Kafka服务中。

[0108] 所述第一预设值大于所述第二预设值,所述第二预设值大于所述第三预设值。通过设置多个预设值对不同下载量建设多个不同的安装渠道,可以更快更准确的实现应用组件的下载。

[0109] 在本实施例中,可选地,将下载所述多个得到的每个组件复制到所述临时安装目录,包括:利用多个更新组件的标识,从服务器处获取所述多个版本的组件的包含该组件的校验信息的索引信息;利用所述校验信息对下载得到多个组件进行校验,在校验通过后,将组件复制到所述临时安装目录。本地组件加载模块,适于加载与本地系统组件库匹配的包含一个或多个组件的扫描库;组件加载确定模块,适于利用加载的扫描库按照扫描规则对本地存储组件进行扫描,得到组件的信息,并确定待安装的组件对应的、且本地系统中未存在的多个组件的标识;本地组件更新模块,适于利用所述多组件的标识从文件服务器处获取所述多个版本的组件文件,进而利用多个组件文件对本地组件库内的组件进行更新、修复。

[0110] 可选地,所述加载模块还适于:获取本地系统的扫描库以及外部的数据发布服务器通过指定通道下发的与本地系统组件库匹配的包含一个或多个不同版本的组件的扫描库;加载获取的所述扫描库。可选地,所述文件服务器包括多个内容分发网络CDN结点,各CDN结点同步有各组件对应的更新文件;服务器建立有更新文件的标识和地址信息的索引。

[0111] 上述实施例中的集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用,可以存储在上述计算机可读的存储介质中。基于这样的理解,本发明的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的全部或部分可以以软件产品的形式体现出来,该计算机软件产品存储在存储介质中,包括若干指令用以使得一台或多台计算机设备(可为个人计算机、服务器或者网络设备)执行本发明各个实施例所述方法的全部或部分步骤。

[0112] 在本发明的上述实施例中,对各个实施例的描述都各有侧重,某个实施例中未详述的部分,可以参见其他实施例的相关描述。

[0113] 在本申请所提供的几个实施例中,应该理解到,所揭露的客户端,可通过其它的方式实现。其中,以上所描述的装置实施例仅仅是示意性的,例如所述单元的划分,仅仅为一种逻辑功能划分,实际实现时可以有另外的划分方式,例如多个单元或组件可以结合或者可以集成到另一个系统,或一些特征可以忽略,或不执行。另一点,所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口,单元或模块的间接耦合或通信连接,可以是电性或其它的形式。

[0114] 所述作为分离部件说明的单元可以是或者也可以不是物理上分开的,作为单元显示的部件可以是或者也可以不是物理单元,即可以位于一个地方,或者也可以分布到多个

网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本实施例方案的目的。

[0115] 另外,在本发明各个实施例中的各功能单元可以集成在一个处理单元中,也可以是各个单元单独物理存在,也可以两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现,也可以采用软件功能单元的形式实现。

[0116] 以上所述仅是本发明的优选实施方式,应当指出,对于本技术领域的普通技术人员来说,在不脱离本发明原理的前提下,还可以做出若干改进和润饰,这些改进和润饰也应视为本发明的保护范围。

[0117] 还需要说明的是,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

[0118] 本领域技术人员应明白,本申请的实施例可提供为方法、系统或计算机程序产品。因此,本申请可采用完全硬件实施例、完全软件实施例或结合软件和硬件方面的实施例的形式。而且,本申请可采用在一个或多个其中包含有计算机可用程序代码的计算机可用存储介质(包括但不限于磁盘存储器、CD-ROM、光学存储器等)上实施的计算机程序产品的形式。

[0119] 虽然上面已经参考各种实施例描述了本发明,但是应当理解,在不脱离本发明的范围的情况下,可以进行许多改变和修改。因此,其旨在上述详细描述被认为是例示性的而非限制性的,并且应当理解,以下权利要求(包括所有等同物)旨在限定本发明的精神和范围。以上这些实施例应理解为仅用于说明本发明而不用于限制本发明的保护范围。在阅读了本发明的记载的内容之后,技术人员可以对本发明作各种改动或修改,这些等效变化和修饰同样落入本发明权利要求所限定的范围。

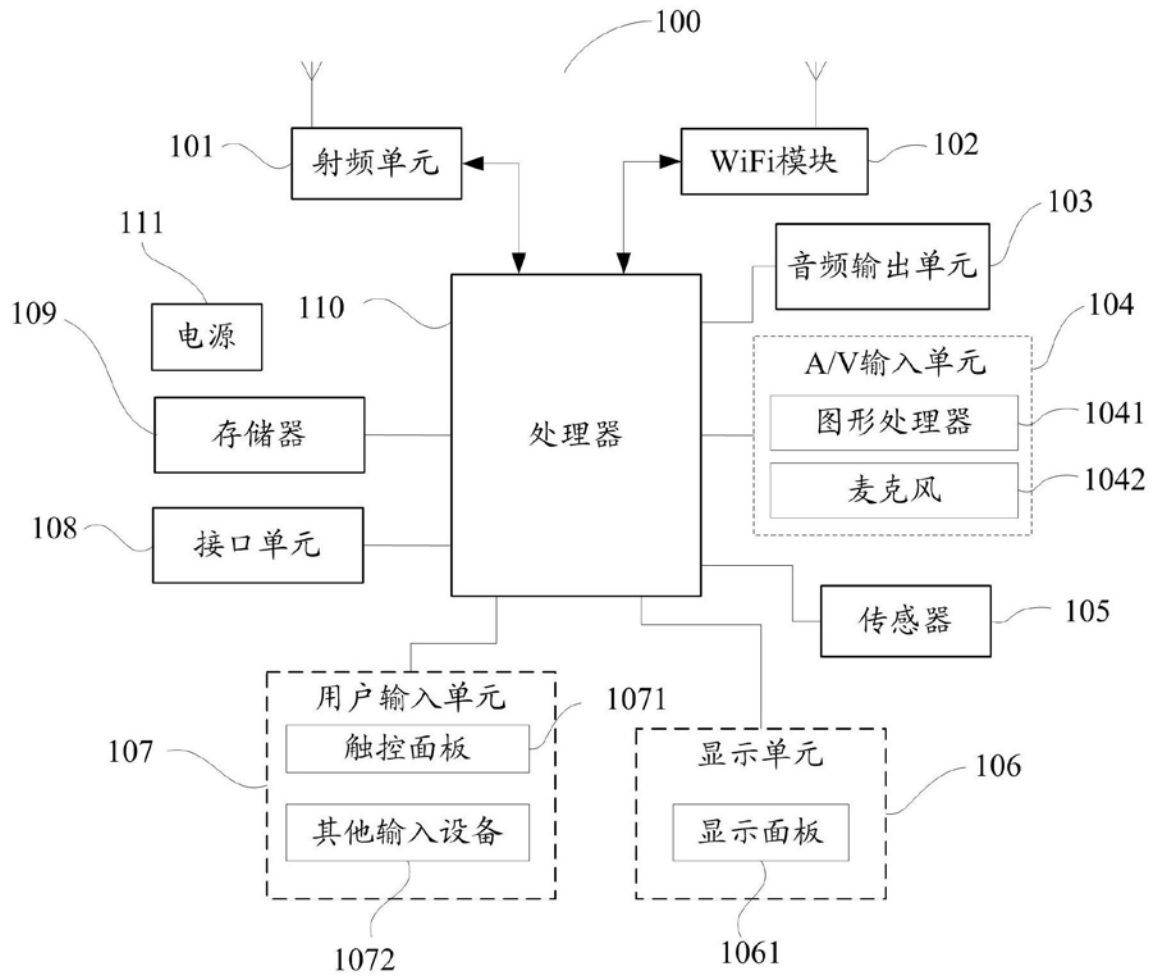


图1

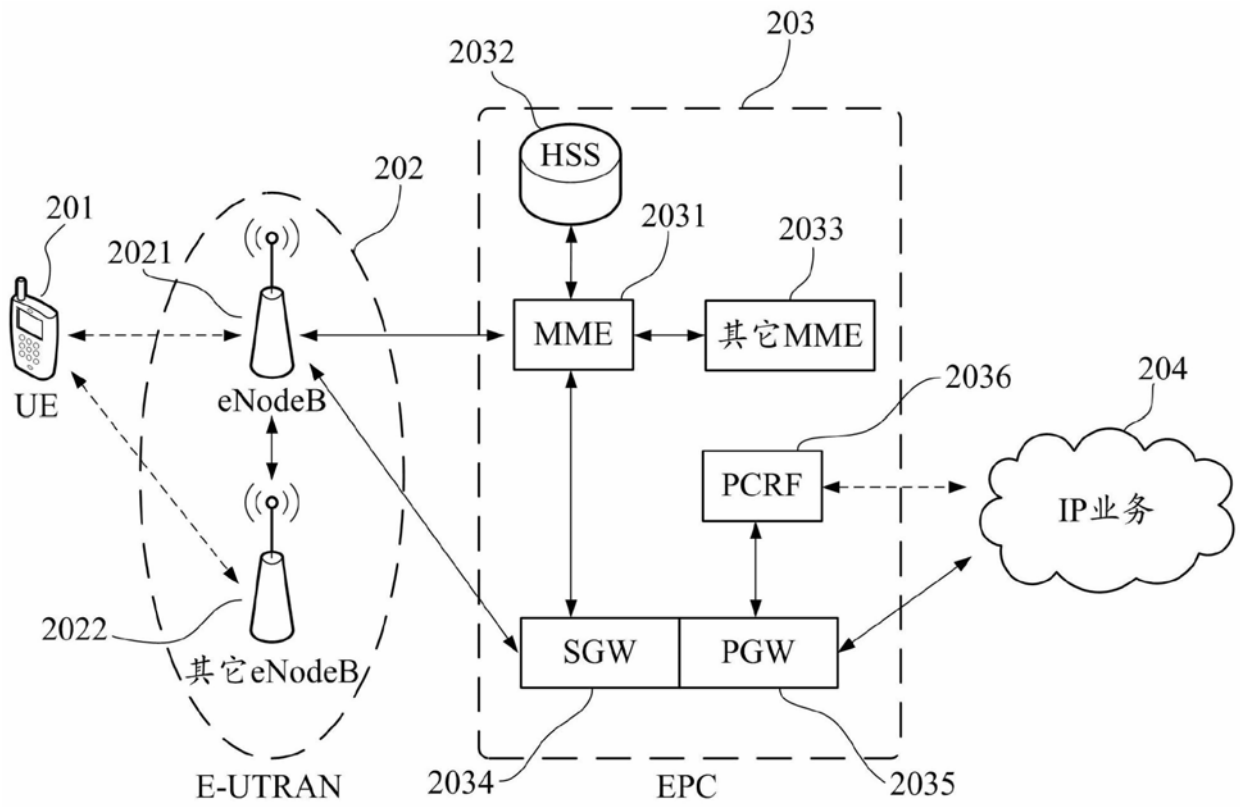


图2

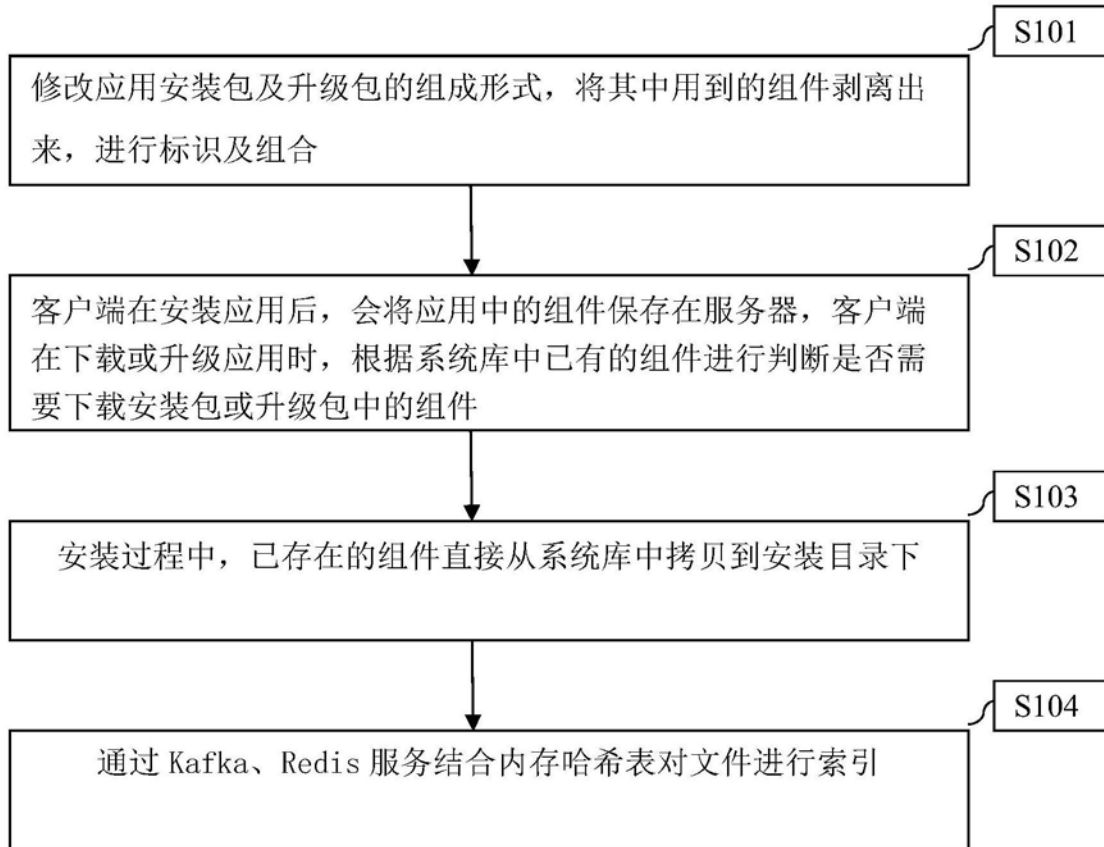


图3

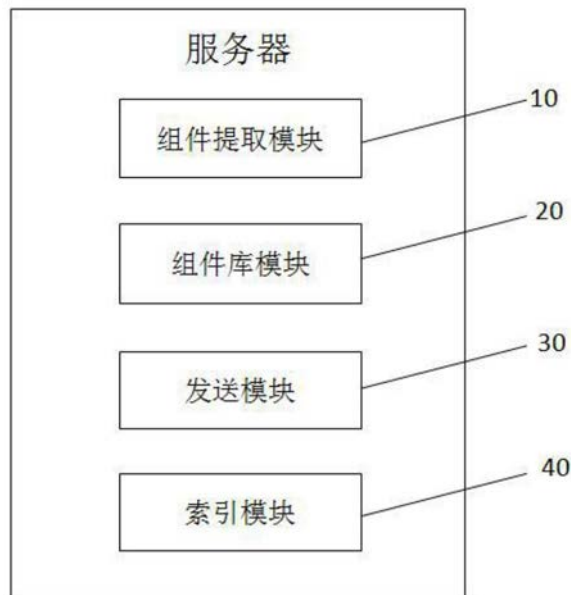


图4

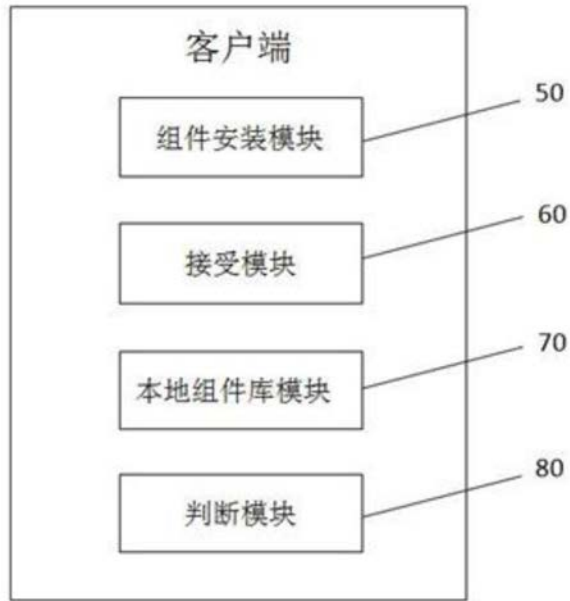


图5