



(12) 发明专利申请

(10) 申请公布号 CN 112256989 A

(43) 申请公布日 2021.01.22

(21) 申请号 202011133908.1

(22) 申请日 2020.10.21

(71) 申请人 平安科技(深圳)有限公司

地址 518000 广东省深圳市福田区福田街
道福安社区益田路5033号平安金融中
心23楼

(72) 发明人 何兵 程达 镐振鹏 李林果
李嘉玲

(74) 专利代理机构 深圳市赛恩倍吉知识产权代
理有限公司 44334

代理人 迟珊珊

(51) Int. Cl.

G06F 16/955 (2019.01)

G06F 16/957 (2019.01)

G06F 16/958 (2019.01)

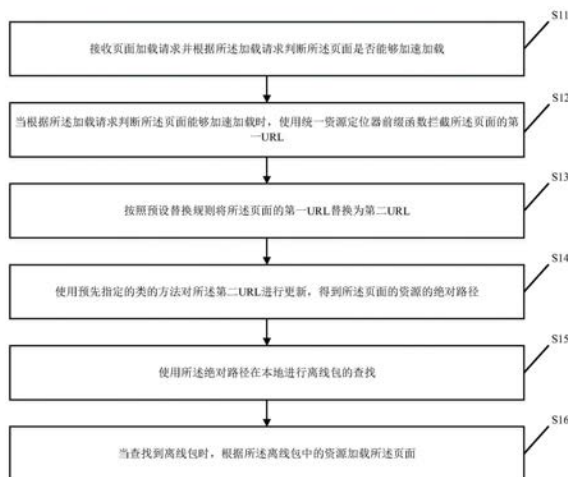
权利要求书2页 说明书11页 附图2页

(54) 发明名称

基于离线包的页面加载方法、装置、终端设备
及存储介质

(57) 摘要

本发明涉及互联网技术领域,提供一种基于
离线包的页面加载方法、装置、终端设备及存储
介质,包括:接收页面加载请求并根据所述加载
请求判断所述页面是否能够加速加载;当根据所
述加载请求判断所述页面能够加速加载时,使用
统一资源定位器前缀函数拦截所述页面的第一
URL;按照预设替换规则将所述页面的第一URL
替换为第二URL;指定的类使用预先指定的类
的方法对所述第二URL进行更新,得到所述页面
的资源绝对路径;使用所述绝对路径在本地进行
离线包的查找;当查找到离线包时,根据所述离
线包中的资源加载所述页面。本发明能够提升
页面加载的效率。



1. 一种基于离线包的页面加载方法,其特征在于,所述方法包括:
 - 接收页面加载请求并根据所述加载请求判断所述页面是否能够加速加载;
 - 当根据所述加载请求判断所述页面能够加速加载时,通过统一资源定位器前缀函数拦截所述页面的第一URL;
 - 按照预设替换规则将所述页面的第一URL替换为第二URL;
 - 使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径;
 - 使用所述绝对路径在本地进行离线包的查找;
 - 当查找到离线包时,根据所述离线包中的资源加载所述页面。
2. 如权利要求1所述的基于离线包的页面加载方法,其特征在于,所述根据所述加载请求判断所述页面是否能够加速加载包括:
 - 解析所述加载请求得到所述页面的参数信息;
 - 识别所述参数信息中的字段是否为预设字段;
 - 当所述参数信息中的字段为所述预设字段时,确定所述页面能够加速加载;
 - 当所述参数信息中的字段不为所述预设字段时,确定所述页面不能够加速加载。
3. 如权利要求1所述的基于离线包的页面加载方法,其特征在于,在所述接收页面加载请求之前,所述方法还包括:
 - 通过指定的接口拉取压缩包的下载地址;
 - 启动下载流程根据所述下载地址下载压缩包;
 - 解压缩所述压缩包到本地沙盒中得到能够加速加载的页面的参数信息及多个资源和资源的路径;
 - 存储所述参数信息及多个资源和资源的路径在预设数据库表中。
4. 如权利要求1所述的基于离线包的页面加载方法,其特征在于,所述按照预设替换规则将所述页面的第一URL替换为第二URL包括:
 - 获取所述指定的类的前缀;
 - 使用所述指定的类的前缀替换所述第一URL中的超文本传输协议得到第二URL。
5. 如权利要求1所述的基于离线包的页面加载方法,其特征在于,所述使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径包括:
 - 在所述指定的类的回调方法中,将所述第二URL的前缀替换为超文本传输协议得到原始URL;
 - 删除所述原始URL中的基础URL得到资源的相对路径;
 - 将所述相对路径与本地沙盒的沙盒路径进行拼接得到资源的绝对路径。
6. 如权利要求1至5中任意一项所述的基于离线包的页面加载方法,其特征在于,所述方法还包括:
 - 计算所述离线包中的资源的第一散列值;
 - 获取应用服务器上对应所述离线包的最新资源并计算所述最新资源的第二散列值;
 - 判断所述第一散列值和第二散列值是否相同;
 - 当所述第一散列值和第二散列值不相同,从所述应用服务器下载所述最新资源;
 - 根据所述最新资源加载所述页面。
7. 如权利要求6所述的基于离线包的页面加载方法,其特征在于,在所述计算所述离线

包中的资源的第一散列值之前,所述方法还包括:判断是否满足资源在线更新的触发条件,所述触发条件包括:应用程序启动、应用程序进入前台、应用程序距离上次更新时间超过预设阈值中的一种或多种;当满足资源在线更新的触发条件时,计算所述离线包中的资源的第一散列值。

8. 一种基于离线包的页面加载装置,其特征在于,所述装置包括:

判断模块,用于接收页面加载请求并根据所述加载请求判断所述页面是否能够加速加载;

注册模块,用于当根据所述加载请求判断所述页面能够加速加载时,使用统一资源定位器前缀函数拦截所述页面的第一URL;

替换模块,用于按照预设替换规则将所述页面的第一URL替换为第二URL;

更新模块,用于使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径;

查找模块,用于使用所述绝对路径在本地进行离线包的查找;

加载模块,用于当查找到离线包时,根据所述离线包中的资源加载所述页面。

9. 一种终端设备,其特征在于,所述终端设备包括处理器,所述处理器用于执行存储器中存储的计算机程序时实现如权利要求1至7中任意一项所述的基于离线包的页面加载方法。

10. 一种计算机可读存储介质,所述计算机可读存储介质上存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现如权利要求1至7中任意一项所述的基于离线包的页面加载方法。

基于离线包的页面加载方法、装置、终端设备及存储介质

技术领域

[0001] 本发明涉及互联网技术领域,具体涉及一种基于离线包的页面加载方法、装置、终端设备及存储介质。

背景技术

[0002] H5 (html5) 是一种超文本标记语言,具有跨平台、开发容易、效率高、方便调试等优点。移动客户端在加载H5页面的时候,容易受网络质量的影响,如果网络质量较差,则会导致加载时间较长或者加载失败。

[0003] 现有技术中,一般是将资源进行压缩,然后优化资源的加载方式,比如可以将显示在屏幕中的图片优先加载,屏幕之外的地方延后加载。然而发明人意识到,对于一些CSS和JS文件资源,必须在全部加载完成后才能正常使用,因此必须优先且一次性全部加载完成,但如果CSS和JS文件较大,依然会发生页面卡住的现象,导致H5页面加载效率低,且还会耗费较多数据流量,影响用户体验。

发明内容

[0004] 鉴于以上内容,有必要提出一种基于离线包的页面加载方法、装置、终端设备及存储介质,能够提升页面加载的效率。

[0005] 本发明的第一方面提供一种基于离线包的页面加载方法,所述方法包括:

[0006] 接收页面加载请求并根据所述加载请求判断所述页面是否能够加速加载;

[0007] 当根据所述加载请求判断所述页面能够加速加载时,使用统一资源定位器前缀函数拦截所述页面的第一URL;

[0008] 按照预设替换规则将所述页面的第一URL替换为第二URL;

[0009] 使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径;

[0010] 使用所述绝对路径在本地进行离线包的查找;

[0011] 当查找到离线包时,根据所述离线包中的资源加载所述页面。

[0012] 可选的,所述根据所述加载请求判断所述页面是否能够加速加载包括:

[0013] 解析所述加载请求得到所述页面的参数信息;

[0014] 识别所述参数信息中的字段是否为预设字段;

[0015] 当所述参数信息中的字段为所述预设字段时,确定所述页面能够加速加载;

[0016] 当所述参数信息中的字段不为所述预设字段时,确定所述页面不能够加速加载。

[0017] 可选的,在所述接收页面加载请求之前,所述方法还包括:

[0018] 通过指定的接口拉取压缩包的下载地址;

[0019] 启动下载流程根据所述下载地址下载压缩包;

[0020] 解压缩所述压缩包到本地沙盒中得到能够加速加载的页面的参数信息及多个资源和资源的路径;

- [0021] 存储所述参数信息及多个资源和资源的路径在预设数据库表中。
- [0022] 可选的,所述按照预设替换规则将所述页面的第一URL替换为第二URL包括:
- [0023] 获取所述指定的类的前缀;
- [0024] 使用所述指定的类的前缀替换所述第一URL中的超文本传输协议得到第二URL。
- [0025] 可选的,所述使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径包括:
- [0026] 在所述指定的类的回调方法中,将所述第二URL的前缀替换为超文本传输协议得到原始URL;
- [0027] 删除所述原始URL中的基础URL得到资源的相对路径;
- [0028] 将所述相对路径与本地沙盒的沙盒路径进行拼接得到资源的绝对路径。
- [0029] 可选的,所述方法还包括:
- [0030] 计算所述离线包中的资源的第一散列值;
- [0031] 获取应用服务器上对应所述离线包的最新资源并计算所述最新资源的第二散列值;
- [0032] 判断所述第一散列值和第二散列值是否相同;
- [0033] 当所述第一散列值和第二散列值不相同,从所述应用服务器下载所述最新资源;
- [0034] 根据所述最新资源加载所述页面。
- [0035] 可选的,在所述计算所述离线包中的资源的第一散列值之前,所述方法还包括:判断是否满足资源在线更新的触发条件,所述触发条件包括:应用程序启动、应用程序进入前台、应用程序距离上次更新时间超过预设阈值中的一种或多种;当满足资源在线更新的触发条件时,计算所述离线包中的资源的第一散列值。
- [0036] 本发明的第二方面提供一种基于离线包的页面加载装置,所述装置包括:
- [0037] 判断模块,用于接收页面加载请求并根据所述加载请求判断所述页面是否能够加速加载;
- [0038] 注册模块,用于当根据所述加载请求判断所述页面能够加速加载时,使用统一资源定位器前缀函数拦截所述页面的第一URL;
- [0039] 替换模块,用于按照预设替换规则将所述页面的第一URL替换为第二URL;
- [0040] 更新模块,用于使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径;
- [0041] 查找模块,用于使用所述绝对路径在本地进行离线包的查找;
- [0042] 加载模块,用于当查找到离线包时,根据所述离线包中的资源加载所述页面。
- [0043] 本发明的第三方面提供一种终端设备,所述终端设备包括处理器,所述处理器用于执行存储器中存储的计算机程序时实现所述的基于离线包的页面加载方法。
- [0044] 本发明的第四方面提供一种计算机可读存储介质,所述计算机可读存储介质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所述的基于离线包的页面加载方法。
- [0045] 综上所述,本发明所述的基于离线包的页面加载方法、装置、终端设备及存储介质,在接收到页面加载请求时,先根据所述加载请求判断所述页面是否能够加速加载;当根

据所述加载请求判断所述页面能够加速加载时,使用统一资源定位器前缀函数拦截所述页面的第一URL;按照预设替换规则将所述页面的第一URL替换为第二URL;使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径;使用所述绝对路径在本地进行离线包的查找并根据所述离线包中的资源加载所述页面,有效解决了现有技术中页面加载时间过长的问题,提升了页面加载的效率;通过将页面的第一URL更新为绝对路径并使用绝对路径在本地查找离线包来实现页面的加载,提高了页面加载的准确性。此外,由于加速加载页面的资源,还能节省数据流量,提高用户体验。

附图说明

[0046] 图1是本发明实施例一提供的基于离线包的页面加载方法的流程图。

[0047] 图2是本发明实施例二提供的基于离线包的页面加载装置的结构图。

[0048] 图3是本发明实施例三提供的终端设备的结构示意图。

具体实施方式

[0049] 为了能够更清楚地理解本发明的上述目的、特征和优点,下面结合附图和具体实施例对本发明进行详细描述。需要说明的是,在不冲突的情况下,本发明的实施例及实施例中的特征可以相互组合。

[0050] 除非另有定义,本文所使用的所有的技术和科学术语与属于本发明的技术领域的技术人员通常理解的含义相同。本文中在本发明的说明书中所使用的术语只是为了描述具体的实施例的目的,不是旨在于限制本发明。

[0051] 本发明实施例提供的基于离线包的页面加载方法由终端设备执行,相应地,基于离线包的页面加载装置运行于终端设备中。

[0052] 图1是本发明实施例一提供的基于离线包的页面加载方法的流程图。所述基于离线包的页面加载方法具体包括以下步骤,根据不同的需求,该流程图中步骤的顺序可以改变,某些可以省略。

[0053] S11,接收页面加载请求并根据所述加载请求判断所述页面是否能够加速加载。

[0054] 当用户需要访问某一页面时,可以通过所持有的终端设备或者终端设备中安装的APP发起页面加载请求,此时终端设备接收到用户发送的页面加载请求,启动对该页面加载请求的处理流程。本实施例中的页面是指H5页面。

[0055] 在一个可选的实施例中,所述根据所述加载请求判断所述页面是否能够加速加载包括:

[0056] 解析所述加载请求得到所述页面的参数信息;

[0057] 识别所述参数信息中的字段是否为预设字段;

[0058] 当所述参数信息中的字段为所述预设字段时,确定所述页面能够加速加载;

[0059] 当所述参数信息中的字段不为所述预设字段时,确定所述页面不能够加速加载。

[0060] 其中,所述资源可以包括:HTML代码,CSS代码,JS代码和静态图片。

[0061] 开发者在开发某个APP时,会根据APP中页面的资源的大小确定页面是否需要加速加载。当某个页面的资源较大(例如,图片大于10M),则标记页面为加速加载。当某个页面的资源较小,则标记页面为普通加载。终端设备在接收到页面加载请求时,通过解析加载请求

可以获取该待加载页面的参数信息(例如,页面名称,页面ID等),终端设备先通过本地预先设置的数据库表识别所述参数信息中的字段是否为预设字段,从而确定所述页面是否能够加速加载。其中,所述预设字段可以为预设名称,预设ID等。

[0062] 在一个可选的实施例中,在确定所述页面不能够加速加载之后,可以按照普通加载方式加载所述页面,即从应用服务器中获取所述页面的资源实现对所述页面的加载。

[0063] 在一个可选的实施例中,在所述接收页面加载请求之前,所述方法还包括:

[0064] 通过指定的接口拉取压缩包的下载地址;

[0065] 启动下载流程根据所述下载地址下载压缩包;

[0066] 解压缩所述压缩包到本地沙盒中得到能够加速加载的页面的参数信息及多个资源和资源的路径;

[0067] 存储所述参数信息及多个资源和资源的路径在预设数据库表中。

[0068] APP服务器通过网络爬虫或者其他手段获取能够加速加载的页面的资源,将能够加速加载的页面的资源压缩为压缩包,并集中存储在统一的文件夹中,其中,所述文件夹的路径与所述能够加速加载的页面的资源的路径一致。

[0069] 终端设备通过指定的接口拉取压缩包的下载地址,可以通过指定的接口的参数进行延后下载,并发量控制等等。其中,所述指定的接口是指终端设备与APP服务器通信连接的接口。

[0070] S12,当根据所述加载请求判断所述页面能够加速加载时,使用统一资源定位器前缀函数拦截所述页面的第一URL。

[0071] 其中,所述统一资源定位器前缀函数可以为URLSchemeHandler函数。

[0072] 终端设备中的APP在安装时可以在系统中注册自己的统一资源定位器前缀函数,方便与其他APP之间相互调用。统一资源定位器前缀函数唯一标识一个APP。具体实施时,首先在配置文件info.plist里添加URL类型,然后在URL类型中添加统一资源定位器前缀。

[0073] 为了避免在加载HTTP协议的网页时出现加载出错的现象,终端设备通过URLSchemeHandler函数对HTTP协议的网页进行拦截。终端设备中的APP在注册自己的统一资源定位器前缀函数时还预先指定一个类。指定的类的前缀为事先定义好的字符串,例如paminawebcache。

[0074] URLSchemeHandler函数的作用是开发者可以将指定的类的前缀以及处理这个前缀的类传递给wkwebview,当wkwebview中的链接、请求、加载资源的URL前面的前缀与指定的前缀一致的时候,系统就会自动将加载与之对应的类的多个回调方法,所述回调方法可以为startURLSchemeTask和stopURLSchemeTask方法。

[0075] S13,按照预设替换规则将所述页面的第一URL替换为第二URL。

[0076] 其中,所述预设替换规则为预先设置的用于将HTTP协议的网页的第一URL替换为非HTTP协议的第二URL。

[0077] 在一个可选的实施例中,所述按照预设替换规则将所述页面的第一URL替换为第二URL包括:

[0078] 获取所述指定的类的前缀;

[0079] 使用所述指定的类的前缀替换所述第一URL中的超文本传输协议得到第二URL。

[0080] 其中,所述超文本传输协议(Hypertext Transfer Protocol over Secure

Socket Layer,HTTPS)是以安全为目标的HTTP通道。示例性的,假设第一URL为https://mina-core-stgl.pingan.com/6525/js/app.99ef97c5.js,按照预设替换规则得到的第二URL为paminawebcache://mina-core-stgl.pingan.com/6525/js/app.99ef97c5.js。

[0081] S14,使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径。

[0082] 终端设备加载第二URL时,当判断到第二URL的前缀为指定的类的前缀时,自动调用指定的类中的startURLSchemeTask方法和stopURLSchemeTask方法。

[0083] 在一个可选的实施例中,所述使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径包括:

[0084] 在所述指定的类的回调方法中,将所述第二URL的前缀替换为超文本传输协议得到原始URL;

[0085] 删除所述原始URL中的基础URL得到资源的相对路径;

[0086] 将所述相对路径与本地沙盒的沙盒路径进行拼接得到资源的绝对路径。

[0087] 所述回调方法可以为startURLSchemeTask方法。

[0088] 由于终端设备的本地硬盘无法识别第二URL,需要通过对第二URL进行更新得到绝对路径,使得本地硬盘能够识别绝对路径,便于在本地中寻找离线包。示例性的,所述原始URL中的基础URL为https://mina-core-stgl.pingan.com/6525,那么在触发startURLSchemeTask方法对所述第二URL进行更新之后,得到所述页面的资源的绝对路径为/js/app.99ef97c5.js。

[0089] S15,使用所述绝对路径在本地进行离线包的查找。

[0090] 终端设备通过绝对路径在本地查找是否有离线包。

[0091] S16,当查找到离线包时,根据所述离线包中的资源加载所述页面。

[0092] 当在本地查找到离线包时,从查找到的所述离线包中获取加载所述待加载页面所需要的资源,并利用所述页面资源执行页面渲染操作以完成页面加载。

[0093] 在一个可选的实施例中,当没有查找到离线包时,根据所述第一URL从应用服务器中获取所述页面的资源实现对所述页面的加载。

[0094] 在一个可选的实施例中,所述方法还可以包括:

[0095] 计算所述离线包中的资源的第一散列值;

[0096] 获取应用服务器上对应所述离线包的最新资源并计算所述最新资源的第二散列值;

[0097] 判断所述第一散列值和第二散列值是否相同;

[0098] 当所述第一散列值和第二散列值不相同,从所述应用服务器下载所述最新资源;

[0099] 根据所述最新资源加载所述页面。

[0100] 该可选的实施例中,可以通过散列函数计算散列值,所述散列函数可以包括,但不限于:消息摘要算法(Message Digest Algorithm,MD5)。具体实施时,将页面的资源作为MD5的输入从而得到固定长度的MD5值。MD5算法不仅可确保一致对比的准确性、安全性,且可大大减少对比计算工作量,进而提升对比速度,提升用户体验。

[0101] 该可选的实施例,通过对离线包的资源的散列值与最新资源的散列值的对比,可

以确定所述页面的资源是否需要更新;如果需要更新,则重新下载资源并打包;如果不需要更新,则不再下载资源。

[0102] 在一个可选的实施例中,在所述计算所述离线包中的资源的第一散列值之前,所述方法还包括:判断是否满足资源在线更新的触发条件。

[0103] 其中,所述触发条件可以包括:应用程序启动、应用程序进入前台、应用程序距离上次更新时间超过预设阈值中的一种或多种。当满足资源在线更新的触发条件时,执行计算所述离线包中的资源的第一散列值。

[0104] 该可选的实施例,通过判断是否满足资源在线更新的触发条件来确定是否更新资源,可保障整个逻辑的通畅性,同时又避免APP运行后错过资源在线更新的触发。通过配备在线更新触发机制,可避免APP过于频繁与APP服务器的验证,既降低APP服务器的压力,也能提升APP的运行流畅性。此外,在非最新情况下,下载最新的资源,能够实现应用程序的本地离线缓存+远程在线更新的目的。

[0105] 综上,本发明在接收到页面加载请求时,先根据所述加载请求判断所述页面是否能够加速加载;当根据所述加载请求判断所述页面能够加速加载时,使用统一资源定位器前缀函数拦截所述页面的第一URL;按照预设替换规则将所述页面的第一URL替换为第二URL;使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径;使用所述绝对路径在本地进行离线包的查找并根据所述离线包中的资源加载所述页面,有效解决了现有技术中页面加载时间过长的问题,提升了页面加载的效率;通过将页面的第一URL更新为绝对路径并使用绝对路径在本地查找离线包来实现页面的加载,提高了页面加载的准确性。此外,由于加速加载页面的资源,还能节省数据流量,提高用户体验。

[0106] 需要强调的是,为进一步保证上述离线包的私密性和安全性,上述离线包可存储于区块链的节点中。

[0107] 图2是本发明实施例二提供的基于离线包的页面加载装置的结构图。

[0108] 在一些实施例中,所述基于离线包的页面加载装置20可以包括多个由计算机程序段所组成的功能模块。所述基于离线包的页面加载装置20中的各个程序段的计算机程序可以存储于终端设备的存储器中,并由至少一个处理器所执行,以执行(详见图1描述)基于离线包的页面加载的功能。

[0109] 本实施例中,所述基于离线包的页面加载装置20根据其所执行的功能,可以被划分为多个功能模块。所述功能模块可以包括:判断模块201、下载模块202、注册模块203、替换模块204、更新模块205、查找模块206、加载模块207及计算模块208。本发明所称的模块是指一种能够被至少一个处理器所执行并且能够完成固定功能的一系列计算机程序段,其存储在存储器中。在本实施例中,关于各模块的功能将在后续的实施例中详述。

[0110] 所述判断模块201,用于接收页面加载请求并根据所述加载请求判断所述页面是否能够加速加载。

[0111] 当用户需要访问某一页面时,可以通过所持有的终端设备或者终端设备中安装的APP发起页面加载请求,此时终端设备接收到用户发送的页面加载请求,启动对该页面加载请求的处理流程。本实施例中的页面是指H5页面。

[0112] 在一个可选的实施例中,所述判断模块201根据所述加载请求判断所述页面是否能够加速加载包括:

- [0113] 解析所述加载请求得到所述页面的参数信息；
- [0114] 识别所述参数信息中的字段是否为预设字段；
- [0115] 当所述参数信息中的字段为所述预设字段时，确定所述页面能够加速加载；
- [0116] 当所述参数信息中的字段不为所述预设字段时，确定所述页面不能够加速加载。
- [0117] 其中，所述资源可以包括：HTML代码，CSS代码，JS代码和静态图片。
- [0118] 开发者在开发某个APP时，会根据APP中页面的资源的大小确定页面是否需要加速加载。当某个页面的资源较大（例如，图片大于10M），则标记页面为加速加载。当某个页面的资源较小，则标记页面为普通加载。终端设备在接收到页面加载请求时，通过解析加载请求可以获得该待加载页面的参数信息（例如，页面名称，页面ID等），终端设备先通过本地预先设置的数据库表识别所述参数信息中的字段是否为预设字段，从而确定所述页面是否能够加速加载。其中，所述预设字段可以为预设名称，预设ID等。
- [0119] 在一个可选的实施例中，在确定所述页面不能够加速加载之后，可以按照普通加载方式加载所述页面，即从应用服务器中获取所述页面的资源实现对所述页面的加载。
- [0120] 所述下载模块202，用于在所述接收页面加载请求之前，下载压缩包。
- [0121] 在一个可选的实施例中，所述下载模块202下载压缩包包括：
- [0122] 通过指定的接口拉取压缩包的下载地址；
- [0123] 启动下载流程根据所述下载地址下载压缩包；
- [0124] 解压缩所述压缩包到本地沙盒中得到能够加速加载的页面的参数信息及多个资源和资源的路径；
- [0125] 存储所述参数信息及多个资源和资源的路径在预设数据库表中。
- [0126] APP服务器通过网络爬虫或者其他手段获取能够加速加载的页面的资源，将能够加速加载的页面的资源压缩为压缩包，并集中存储在统一的文件夹中，其中，所述文件夹的路径与所述能够加速加载的页面的资源的路径一致。
- [0127] 终端设备通过指定的接口拉取压缩包的下载地址，可以通过指定的接口的参数进行延后下载，并发量控制等等。其中，所述指定的接口是指终端设备与APP服务器通信连接的接口。
- [0128] 所述注册模块203，用于当根据所述加载请求判断所述页面能够加速加载时，使用统一资源定位器前缀函数拦截所述页面的第一URL。
- [0129] 其中，所述统一资源定位器前缀函数可以为URLSchemeHandler函数。
- [0130] 终端设备中的APP在安装时可以在系统中注册自己的统一资源定位器前缀函数，方便与其他APP之间相互调用。统一资源定位器前缀函数唯一标识一个APP。具体实施时，首先在配置文件info.plist里添加URL类型，然后在URL类型中添加统一资源定位器前缀。
- [0131] 为了避免在加载HTTP协议的网页时出现加载出错的现象，终端设备通过URLSchemeHandler函数对HTTP协议的网页进行拦截。终端设备中的APP在注册自己的统一资源定位器前缀函数时还预先指定一个类。指定的类的前缀为事先定义好的字符串，例如paminawebcache。
- [0132] URLSchemeHandler函数的作用是开发者可以将指定的类的前缀以及处理这个前缀的类传递给wkwebview，当wkwebview中的链接、请求、加载资源的URL前面的前缀与指定的前缀一致的时候，系统就会自动将加载与之对应的类的多个回调方法，所述回调方法可以

为startURLSchemeTask和stopURLSchemeTask方法。

[0133] 所述替换模块204,用于按照预设替换规则将所述页面的第一URL替换为第二URL。

[0134] 其中,所述预设替换规则为预先设置的用于将HTTP协议的网页的第一URL替换为非HTTP协议的第二URL。

[0135] 在一个可选的实施例中,所述替换模块204按照预设替换规则将所述页面的第一URL替换为第二URL包括:

[0136] 获取所述指定的类的前缀;

[0137] 使用所述指定的类的前缀替换所述第一URL中的超文本传输协议得到第二URL。

[0138] 其中,所述超文本传输协议(Hypertext Transfer Protocol over Secure Socket Layer,HTTPS)是以安全为目标的HTTP通道。示例性的,假设第一URL为https://mina-core-stg1.pingan.com/6525/js/app.99ef97c5.js,按照预设替换规则得到的第二URL为paminawebcache://mina-core-stg1.pingan.com/6525/js/app.99ef97c5.js。

[0139] 所述更新模块205,用于使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径。

[0140] 终端设备加载第二URL时,当判断到第二URL的前缀为指定的类的前缀时,自动调用指定的类中的startURLSchemeTask方法和stopURLSchemeTask方法。

[0141] 在一个可选的实施例中,所述更新模块205使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径包括:

[0142] 在所述指定的类的回调方法中,将所述第二URL的前缀替换为超文本传输协议得到原始URL;

[0143] 删除所述原始URL中的基础URL得到资源的相对路径;

[0144] 将所述相对路径与本地沙盒的沙盒路径进行拼接得到资源的绝对路径。

[0145] 所述回调方法可以为startURLSchemeTask方法。

[0146] 由于终端设备的本地硬盘无法识别第二URL,需要通过对第二URL进行更新得到绝对路径,使得本地硬盘能够识别绝对路径,便于在本地中寻找离线包。示例性的,所述原始URL中的基础URL为https://mina-core-stg1.pingan.com/6525,那么在触发startURLSchemeTask方法对所述第二URL进行更新之后,得到所述页面的资源的绝对路径为/js/app.99ef97c5.js。

[0147] 所述查找模块206,用于使用所述绝对路径在本地进行离线包的查找。

[0148] 终端设备通过绝对路径在本地查找是否有离线包。

[0149] 所述加载模块207,用于当查找到离线包时,根据所述离线包中的资源加载所述页面。

[0150] 当在本地查找到离线包时,从查找到的所述离线包中获取加载所述待加载页面所需要的资源,并利用所述页面资源执行页面渲染操作以完成页面加载。

[0151] 在一个可选的实施例中,当没有查找到离线包时,根据所述第一URL从应用服务器中获取所述页面的资源实现对所述页面的加载。

[0152] 所述计算模块208,用于计算所述离线包中的资源的第一散列值;获取应用服务器上对应所述离线包的最新资源并计算所述最新资源的第二散列值。

[0153] 所述判断模块201,还用于判断所述第一散列值和第二散列值是否相同。

[0154] 所述下载模块202,还用于当所述第一散列值和第二散列值不相同,从所述应用服务器下载所述最新资源。

[0155] 所述加载模块207,还用于根据所述最新资源加载所述页面。

[0156] 该可选的实施例中,可以通过散列函数计算散列值,所述散列函数可以包括,但不限于:消息摘要算法(Message Digest Algorithm,MD5)。具体实施时,将页面的资源作为MD5的输入从而得到固定长度的MD5值。MD5算法不仅可确保一致对比的准确性、安全性,且可大大减少对比计算工作量,进而提升对比速度,提升用户体验。

[0157] 该可选的实施例,通过对离线包的资源的散列值与最新资源的散列值的对比,可以确定所述页面的资源是否需要更新;如果需要更新,则重新下载资源并打包;如果不需要更新,则不再下载资源。

[0158] 在一个可选的实施例中,所述判断模块201,还用于在所述计算所述离线包中的资源的第一散列值之前,判断是否满足资源在线更新的触发条件。

[0159] 其中,所述触发条件可以包括:应用程序启动、应用程序进入前台、应用程序距离上次更新时间超过预设阈值中的一种或多种。当满足资源在线更新的触发条件时,执行计算所述离线包中的资源的第一散列值。

[0160] 该可选的实施例,通过判断是否满足资源在线更新的触发条件来确定是否更新资源,可保障整个逻辑的通畅性,同时又避免APP运行后错过资源在线更新的触发。通过配备在线更新触发机制,可避免APP过于频繁与APP服务器的验证,既降低APP服务器的压力,也能提升APP的运行流畅性。此外,在非最新情况下,下载最新的资源,能够实现应用程序的本地离线缓存+远程在线更新的目的。

[0161] 综上,本发明在接收到页面加载请求时,先根据所述加载请求判断所述页面是否能够加速加载;当根据所述加载请求判断所述页面能够加速加载时,使用统一资源定位器前缀函数拦截所述页面的第一URL;按照预设替换规则将所述页面的第一URL替换为第二URL;使用预先指定的类的方法对所述第二URL进行更新,得到所述页面的资源的绝对路径;使用所述绝对路径在本地进行离线包的查找并根据所述离线包中的资源加载所述页面,有效解决了现有技术中页面加载时间过长的问题,提升了页面加载的效率;通过将页面的第一URL更新为绝对路径并使用绝对路径在本地查找离线包来实现页面的加载,提高了页面加载的准确性。此外,由于加速加载页面的资源,还能节省数据流量,提高用户体验。

[0162] 需要强调的是,为进一步保证上述离线包的私密性和安全性,上述离线包可存储于区块链的节点中。

[0163] 参阅图3所示,为本发明实施例三提供的终端设备的结构示意图。在本发明较佳实施例中,所述终端设备3包括存储器31、至少一个处理器32、至少一条通信总线33及收发器34。

[0164] 本领域技术人员应该了解,图3示出的终端设备的结构并不构成本发明实施例的限定,既可以是总线型结构,也可以是星形结构,所述终端设备3还可以包括比图示更多或更少的其他硬件或者软件,或者不同的部件布置。

[0165] 在一些实施例中,所述终端设备3是一种能够按照事先设定或存储的指令,自动进行数值计算和/或信息处理的终端设备,其硬件包括但不限于微处理器、专用集成电路、可编程门阵列、数字处理器及嵌入式设备等。所述终端设备3还可包括客户设备,所述客户设

备包括但不限于任何一种可与客户通过键盘、鼠标、遥控器、触摸板或声控设备等方式进行人机交互的电子产品,例如,个人计算机、平板电脑、智能手机、数码相机等。

[0166] 需要说明的是,所述终端设备3仅为举例,其他现有的或今后可能出现的电子产品如可适应于本发明,也应包含在本发明的保护范围以内,并以引用方式包含于此。

[0167] 在一些实施例中,所述存储器31中存储有计算机程序,所述计算机程序被所述至少一个处理器32执行时实现如所述的基于离线包的页面加载方法中的全部或者部分步骤。所述存储器31包括只读存储器(Read-Only Memory,ROM)、可编程只读存储器(Programmable Read-Only Memory,PROM)、可擦除可编程只读存储器(Erasable Programmable Read-Only Memory,EPRM)、一次可编程只读存储器(One-time Programmable Read-Only Memory,OTPROM)、电子擦除式可复写只读存储器(Electrically-Erasable Programmable Read-Only Memory,EEPROM)、只读光盘(Compact Disc Read-Only Memory,CD-ROM)或其他光盘存储器、磁盘存储器、磁带存储器、或者能够用于携带或存储数据的计算机可读的任何其他介质。

[0168] 进一步地,所述计算机可读存储介质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等;存储数据区可存储根据区块链节点的使用所创建的数据等。

[0169] 本发明所指区块链是分布式数据存储、点对点传输、共识机制、加密算法等计算机技术的新型应用模式。区块链(Blockchain),本质上是一个去中心化的数据库,是一串使用密码学方法相关联产生的数据块,每一个数据块中包含了一批网络交易的信息,用于验证其信息的有效性(防伪)和生成下一个区块。区块链可以包括区块链底层平台、平台产品服务层以及应用服务层等。

[0170] 在一些实施例中,所述至少一个处理器32是所述终端设备3的控制核心(Control Unit),利用各种接口和线路连接整个终端设备3的各个部件,通过运行或执行存储在所述存储器31内的程序或者模块,以及调用存储在所述存储器31内的数据,以执行终端设备3的各种功能和处理数据。例如,所述至少一个处理器32执行所述存储器中存储的计算机程序时实现本发明实施例中所述的基于离线包的页面加载方法的全部或者部分步骤;或者实现基于离线包的页面加载装置的全部或者部分功能。所述至少一个处理器32可以由集成电路组成,例如可以由单个封装的集成电路所组成,也可以是由多个相同功能或不同功能封装的集成电路所组成,包括一个或者多个中央处理器(Central Processing unit,CPU)、微处理器、数字处理芯片、图形处理器及各种控制芯片的组合等。

[0171] 在一些实施例中,所述至少一条通信总线33被设置为实现所述存储器31以及所述至少一个处理器32等之间的连接通信。

[0172] 尽管未示出,所述终端设备3还可以包括给各个部件供电的电源(比如电池),优选的,电源可以通过电源管理装置与所述至少一个处理器32逻辑相连,从而通过电源管理装置实现管理充电、放电、以及功耗管理等功能。电源还可以包括一个或一个以上的直流或交流电源、再充电装置、电源故障检测电路、电源转换器或者逆变器、电源状态指示器等任意组件。所述终端设备3还可以包括多种传感器、蓝牙模块、Wi-Fi模块等,在此不再赘述。

[0173] 上述以软件功能模块的形式实现的集成的单元,可以存储在一个计算机可读存储介质中。上述软件功能模块存储在一个存储介质中,包括若干指令用以使得一台计算机

设备(可以是个人计算机,终端设备,或者网络设备等)或处理器(processor)执行本发明各个实施例所述方法的部分。

[0174] 在本发明所提供的几个实施例中,应该理解到,所揭露的装置和方法,可以通过其它的方式实现。例如,以上所描述的装置实施例仅仅是示意性的,例如,所述模块的划分,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。

[0175] 所述作为分离部件说明的模块可以是或者也可以不是物理上分开的,作为模块显示的部件可以是或者也可以不是物理单元,既可以位于一个地方,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模块来实现本实施例方案的目的。

[0176] 另外,在本发明各个实施例中的各功能模块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,也可以采用硬件加软件功能模块的形式实现。

[0177] 对于本领域技术人员而言,显然本发明不限于上述示范性实施例的细节,而且在不背离本发明的精神或基本特征的情况下,能够以其他的具体形式实现本发明。因此,无论从哪一点来看,均应将实施例看作是示范性的,而且是非限制性的,本发明的范围由所附权利要求而不是上述说明限定,因此旨在将落在权利要求的等同要件的含义和范围内的所有变化涵括在本发明内。不应将权利要求中的任何附图标记视为限制所涉及的权利要求。此外,显然“包括”一词不排除其他单元或,单数不排除复数。装置权利要求中陈述的多个单元或装置也可以由一个单元或装置通过软件或者硬件来实现。第一,第二等词语用来表示名称,而并不表示任何特定的顺序。

[0178] 最后应说明的是,以上实施例仅用以说明本发明的技术方案而非限制,尽管参照较佳实施例对本发明进行了详细说明,本领域的普通技术人员应当理解,可以对本发明的技术方案进行修改或等同替换,而不脱离本发明技术方案的精神和范围。

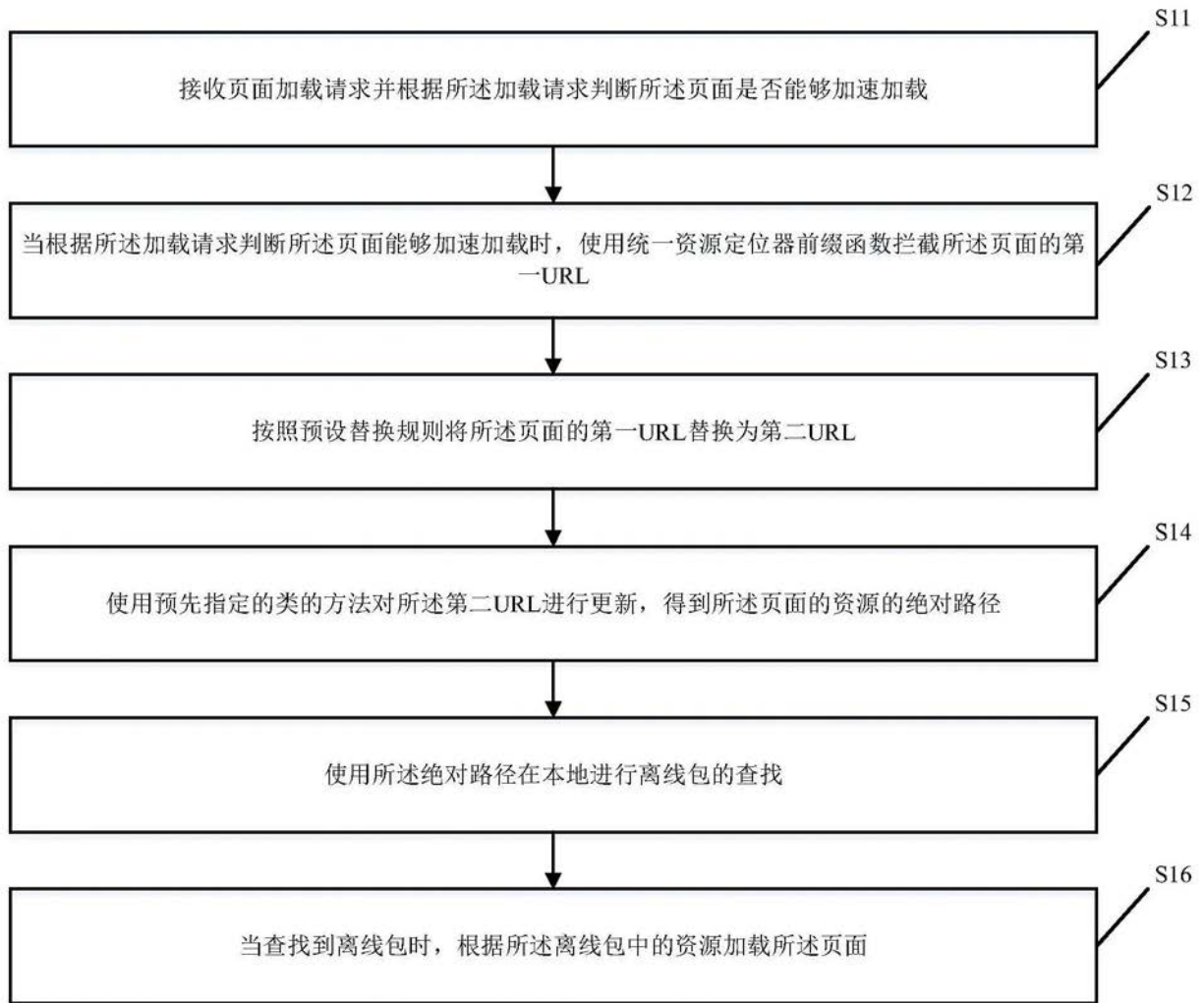


图1

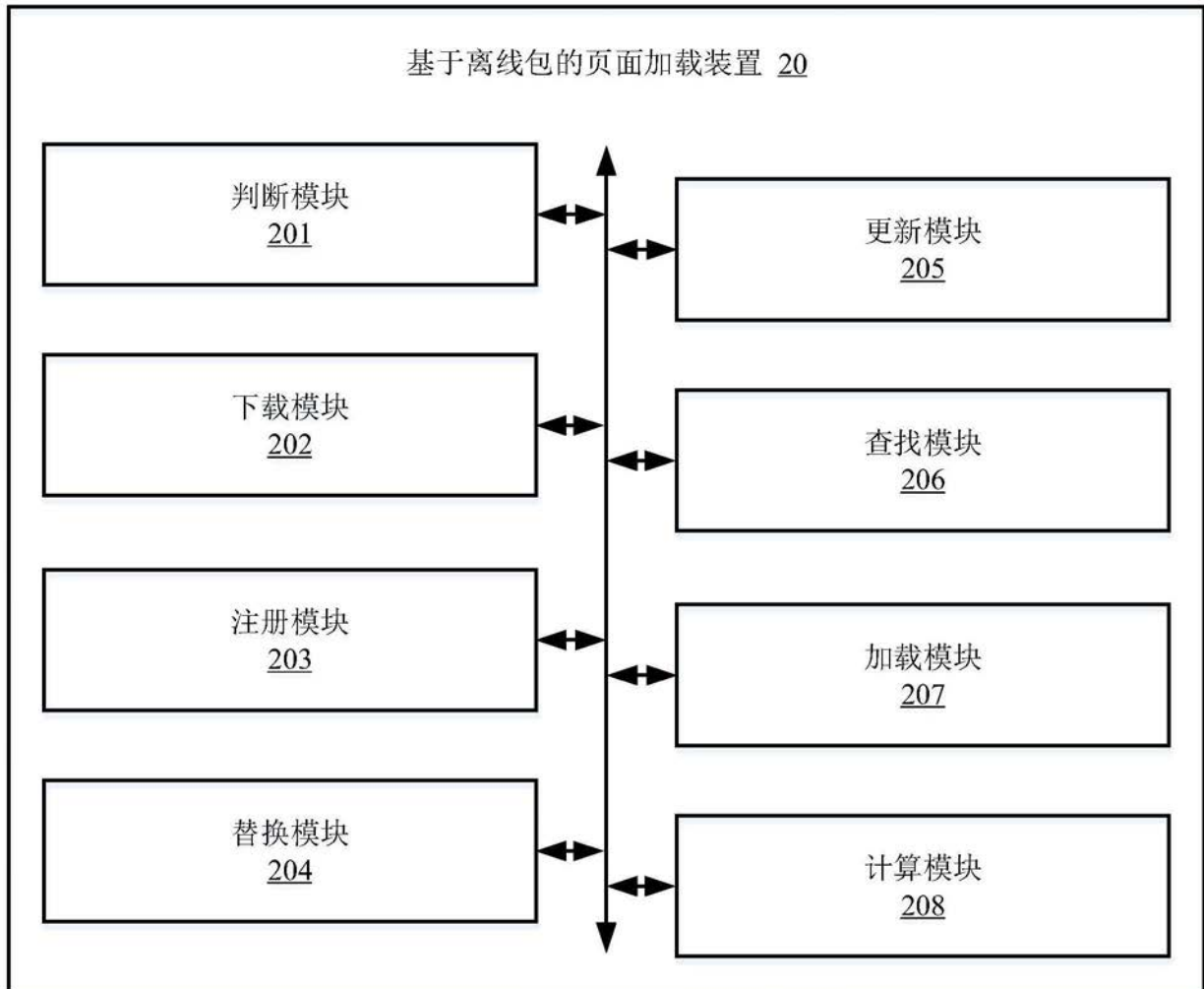


图2

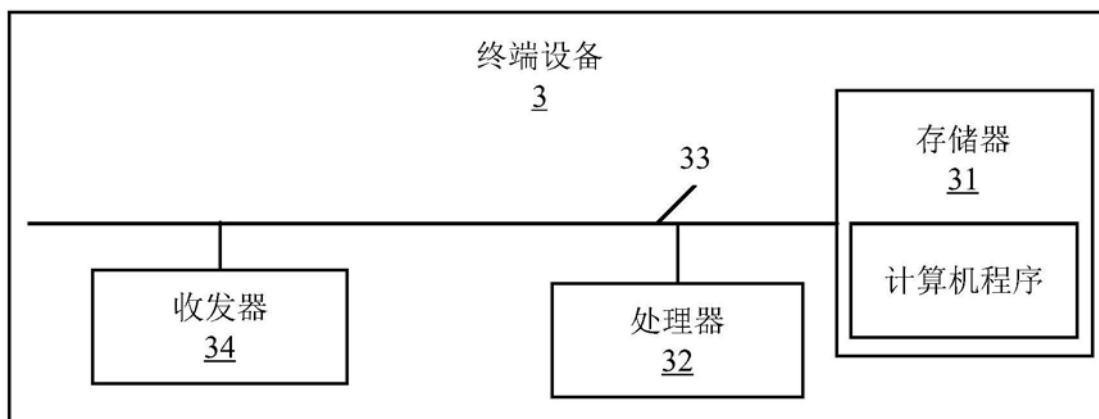


图3