



(12) 发明专利

(10) 授权公告号 CN 110096295 B

(45) 授权公告日 2023. 08. 08

(21) 申请号 201910379464.0

(22) 申请日 2019.05.08

(65) 同一申请的已公布的文献号
申请公布号 CN 110096295 A

(43) 申请公布日 2019.08.06

(73) 专利权人 吉旗(成都)科技有限公司
地址 610000 四川省成都市天府新区天府大道南段846号

(72) 发明人 赖雪菲 李响

(74) 专利代理机构 北京弘权知识产权代理有限公司 11363
专利代理师 逯长明 许伟群

(51) Int. Cl.
G06F 8/65 (2018.01)

(56) 对比文件
CN 109032638 A, 2018.12.18
CN 102946442 A, 2013.02.27

CN 102681825 A, 2012.09.19

CN 103944958 A, 2014.07.23

CN 104133695 A, 2014.11.05

CN 104735152 A, 2015.06.24

CN 106209926 A, 2016.12.07

CN 106657367 A, 2017.05.10

US 2008162674 A1, 2008.07.03

CN 107729041 A, 2018.02.23

CN 109445833 A, 2019.03.08

CN 108304207 A, 2018.07.20

CN 107861739 A, 2018.03.30

CN 108733400 A, 2018.11.02

CN 109656611 A, 2019.04.19

CN 107509086 A, 2017.12.22

张鑫.Android热更新管理系统的研究与设计.《中国优秀硕士学位论文全文数据库信息科技辑》.2019, (第2期), 第1138-671页.

审查员 黄琦

权利要求书3页 说明书8页 附图6页

(54) 发明名称

基于ReactNative的多模块移动应用的热更新方法及系统

(57) 摘要

本发明公开一种基于ReactNative的多模块移动应用的热更新方法及系统。在该方法中,客户端可针对不同ReactNative模块,利用本地的缓存文件或者内置文件进行热更新,即使网络等原因造成模块资源获取失败,也可以获取缓存中最近的资源文件进行界面数据渲染,服务器端可以根据客户端的请求,精准地为模块下发更新资源,能够更加快速的进行App的版本更新、及时解决线上问题,让用户不用重新下载新版本就能体验新版本App的功能。



1. 一种基于ReactNative的多模块移动应用的热更新方法,应用于客户端,其特征在于,所述方法包括如下步骤:

进入移动应用的ReactNative模块;

加载所述移动应用的jsbundle文件和资源文件;

根据所述jsbundle文件和资源文件,向服务器请求所述ReactNative模块的热更新配置信息,所述热更新配置信息包括以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值;

接收服务器返回的热更新配置信息;

根据所述热更新配置信息进行热更新资源的下载;

解压所述热更新资源;

将解压后的所述热更新资源缓存到所述移动应用的沙盒路径中;

加载所述移动应用的jsbundle文件和资源文件包括:

检测所述移动应用的沙盒中是否缓存有所述ReactNative模块的jsbundle文件和资源文件;如果所述移动应用的沙盒中缓存有所述ReactNative模块的jsbundle文件和资源文件,加载缓存的jsbundle文件和资源文件;

如果所述移动应用的沙盒中没有缓存所述ReactNative模块的jsbundle文件和资源文件,加载所述移动应用中内置的jsbundle文件和资源文件;

加载缓存的jsbundle文件和资源文件包括:

将所述ReactNative模块的文件拷贝到currentReadedModule文件中;

返回拷贝后的文件路径;

读取所述拷贝后的文件路径中的jsbundle文件和资源文件包括进行界面渲染;

将解压后的所述热更新资源缓存到所述移动应用的沙盒路径中之后,所述方法还包括:识别所述ReactNative模块是否需要强制更新;

如果所述ReactNative模块需要强制更新,刷新所述ReactNative模块

刷新所述ReactNative模块包括:

向服务器上传刷新页面的请求,所述刷新页面的请求中携带有请求参数,以使得服务器根据所述请求参数查询出未刷新期间产生的数据和补传数据,判断所述未刷新期间产生的数据和补传数据的数据量是否超出预设的最大数据量,在所述未刷新期间产生的数据和补传数据的数据量超出预设的最大数据量时,向客户端返回数据并在返回的数据中携带还有更多数据的标识;

接收服务器返回的数据;

根据服务器返回的数据中的标识,判断是否有更多数据未传回;

如果有更多数据未传回,根据服务器返回的数据的最早产生时间,移除本地缓存中所有比所述最早产生时间早的数据;

将其余的数据缓存于本地数据库中;

刷新显示界面;

所述方法还包括:

向服务器上传加载更多数据的请求,所述加载更多数据的请求中携带有客户端当前界面显示数据中的最早发生时间,以使得服务器查找出比所述最早发生时间更早的数据,将

比所述最早发生时间更早的数据返回至客户端；

接收服务器返回的比所述最早发生时间更早的数据；

将比所述最早发生时间更早的数据缓存本地数据库中；

刷新显示界面。

2. 一种基于ReactNative的多模块移动应用的热更新方法,应用于服务器,其特征在于,所述方法包括如下步骤:

接收客户端上传的对移动应用ReactNative模块的配置信息进行热更新的请求,所述热更新的请求中携带以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值;

根据所述模块唯一标识,查询服务器的数据库中启用的最新资源;

识别是否存在所述最新资源;

如果不存在所述最新资源,向客户端返回请求失败信息;

如果存在所述最新资源,判断所述请求中携带的应用版本号是否大于或等于服务器的数据库中当前配置的应用版本号;

如果所述请求中携带的应用版本号大于或等于服务器的数据库中当前配置的应用版本号,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新;

如果所述请求中携带的应用版本号不大于或等于服务器的数据库中当前配置的应用版本号,判断所述请求中携带的jsbundle版本号是否大于或等于服务器的数据库中当前配置的jsbundle版本号;

如果所述请求中携带的jsbundle版本号大于或等于服务器的数据库中当前配置的jsbundle版本号,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新;

如果所述请求中携带的jsbundle版本号不大于或等于服务器的数据库中当前配置的jsbundle版本号,判断所述请求中携带的jsbundle的Md5值是否大于或等于服务器的数据库中当前配置的jsbundle的Md5值;

如果所述请求中携带的jsbundle的Md5值大于或等于数据库中当前配置的jsbundle的Md5值,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新;

如果所述请求中携带的jsbundle的Md5值不大于或等于服务器的数据库中当前配置的jsbundle的Md5值,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端不对移动应用ReactNative模块的配置信息进行热更新。

3. 一种基于ReactNative的多模块移动应用的热更新系统,其特征在于,所述系统包括客户端和服务器;

所述客户端,用于进入移动应用的ReactNative模块,加载所述移动应用的jsbundle文件和资源文件,根据所述jsbundle文件和资源文件,向服务器请求所述ReactNative模块的热更新配置信息,所述热更新配置信息包括以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值,接收服务器返回的热更新配置信息,根据所述热更新配置信息进行热更新资源的下载,解压所述热更新资源,将解压后的所述热更新资源

缓存到所述移动应用的沙盒路径中；

所述服务器,用于接收客户端上传的对移动应用ReactNative模块的配置信息进行热更新的请求,所述热更新的请求中携带以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值,根据所述模块唯一标识,查询服务器的数据库中启用的最新资源,识别是否存在所述最新资源,如果不存在所述最新资源,向客户端返回请求失败信息,如果存在所述最新资源,判断所述请求中携带的应用版本号是否大于或等于服务器的数据库中当前配置的应用版本号,如果所述请求中携带的应用版本号大于或等于服务器的数据库中当前配置的应用版本号,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新。

基于ReactNative的多模块移动应用的热更新方法及系统

技术领域

[0001] 本发明涉及网络技术领域,尤其涉及一种基于ReactNative的多模块移动应用的热更新方法及系统。

背景技术

[0002] 移动互联网就是将移动通信和互联网二者结合起来成为一体,是指互联网的技术、平台、商业模式和应用与移动通信技术结合并实践的活动的总称。在当今的移动互联网时代,移动应用几乎面临着和Web应用一样的问题,即不跨平台、版本收敛慢、无法及时修复线上问题、需求迭代缓慢等。但移动应用还不能完全转成Web应用,所以App的动态化就成了一种趋势。App的动态化的方案有很多种,ReactNative则是很具有代表性的一种。为了更加快速的进行App的版本更新、及时解决线上问题,让用户不用重新下载新版本就能体验新版本App的功能,亟需提出一种基于ReactNative的多模块移动应用的热更新方法及系统。

发明内容

[0003] 本发明提供一种基于ReactNative的多模块移动应用的热更新方法及系统,能够更加快速的进行App的版本更新、及时解决线上问题,让用户不用重新下载新版本就能体验新版本App的功能。

[0004] 第一方面,本发明提供一种基于ReactNative的多模块移动应用的热更新方法,应用于客户端,所述方法包括如下步骤:

[0005] 进入移动应用的ReactNative模块;

[0006] 加载所述移动应用的jsbundle文件和资源文件;

[0007] 根据所述jsbundle文件和资源文件,向服务器请求所述ReactNative模块的热更新配置信息,所述热更新配置信息包括以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值;

[0008] 接收服务器返回的热更新配置信息;

[0009] 根据所述热更新配置信息进行热更新资源的下载;

[0010] 解压所述热更新资源;

[0011] 将解压后的所述热更新资源缓存到所述移动应用的沙盒路径中。

[0012] 结合第一方面,在第一方面的第一种可实现方式中,加载所述移动应用的jsbundle文件和资源文件包括:

[0013] 检测所述移动应用的沙盒中是否缓存有所述ReactNative模块的jsbundle文件和资源文件;

[0014] 如果所述移动应用的沙盒中缓存有所述ReactNative模块的jsbundle文件和资源文件,加载缓存的jsbundle文件和资源文件;

[0015] 如果所述移动应用的沙盒中没有缓存所述ReactNative模块的jsbundle文件和资源文件,加载所述移动应用中内置的jsbundle文件和资源文件。

[0016] 结合第一方面,在第一方面的第二种可实现方式中,加载缓存的jsbundle文件和资源文件包括:

[0017] 将所述ReactNative模块的文件拷贝到currentReadedModule文件中;

[0018] 返回拷贝后的文件路径;

[0019] 读取所述拷贝后的文件路径中的jsbundle文件和资源文件包括进行界面渲染。

[0020] 结合第一方面,在第一方面的第三种可实现方式中,将解压后的所述热更新资源缓存到所述移动应用的沙盒路径中之后,所述方法还包括:

[0021] 识别所述ReactNative模块是否需要强制更新;

[0022] 如果所述ReactNative模块需要强制更新,刷新所述ReactNative模块。

[0023] 结合第一方面的第三种可实现方式,在第一方面的第四种可实现方式中,刷新所述ReactNative模块包括:

[0024] 向服务器上传刷新页面的请求,所述刷新页面的请求中携带有请求参数,以使得服务器根据所述请求参数查询出未刷新期间产生的数据和补传数据,判断所述未刷新期间产生的数据和补传数据的数据量是否超出预设的最大数据量,在所述未刷新期间产生的数据和补传数据的数据量超出预设的最大数据量时,向客户端返回数据并在返回的数据中携带还有更多数据的标识;

[0025] 接收服务器返回的数据;

[0026] 根据服务器返回的数据中的标识,判断是否有更多数据未传回;

[0027] 如果有更多数据未传回,根据服务器返回的数据的最早产生时间,移除本地缓存中所有比所述最早产生时间早的数据;

[0028] 将其余的数据缓存于本地数据库中;

[0029] 刷新显示界面。

[0030] 结合第一方面的第四种可实现方式,在第一方面的第五种可实现方式中,所述方法还包括:

[0031] 向服务器上传加载更多数据的请求,所述加载更多数据的请求中携带有客户端当前界面显示数据中的最早发生时间,以使得服务器查找出比所述最早发生时间更早的数据,将比所述最早发生时间更早的数据返回至客户端;

[0032] 接收服务器返回的比所述最早发生时间更早的数据;

[0033] 将比所述最早发生时间更早的数据缓存本地数据库中;

[0034] 刷新显示界面。

[0035] 第二方面,本发明提供一种基于ReactNative的多模块移动应用的热更新方法,应用于服务器,所述方法包括如下步骤:

[0036] 接收客户端上传的对移动应用ReactNative模块的配置信息进行热更新的请求,所述热更新的请求中携带以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值;

[0037] 根据所述模块唯一标识,查询服务器的数据库中启用的最新资源;

[0038] 识别是否存在所述最新资源;

[0039] 如果不存在所述最新资源,向客户端返回请求失败信息;

[0040] 如果存在所述最新资源,判断所述请求中携带的应用版本号是否大于或等于服务

器的数据库中当前配置的应用版本号；

[0041] 如果所述请求中携带的应用版本号大于或等于服务器的数据库中当前配置的应用版本号,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新。

[0042] 结合第二方面,在第二方面的第一种可实现方式中,如果所述请求中携带的应用版本号不大于或等于服务器的数据库中当前配置的应用版本号,判断所述请求中携带的jsbundle版本号是否大于或等于服务器的数据库中当前配置的jsbundle版本号;

[0043] 如果所述请求中携带的jsbundle版本号大于或等于服务器的数据库中当前配置的jsbundle版本号,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新。

[0044] 结合第二方面的第一种可实现方式,在第二方面的第二种可实现方式中,如果所述请求中携带的jsbundle版本号不大于或等于服务器的数据库中当前配置的jsbundle版本号,判断所述请求中携带的jsbundle的Md5值是否大于或等于服务器的数据库中当前配置的jsbundle的Md5;

[0045] 如果所述请求中携带的jsbundle的Md5值大于或等于数据库中当前配置的jsbundle的Md5值,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新;

[0046] 如果所述请求中携带的jsbundle的Md5值不大于或等于服务器的数据库中当前配置的jsbundle的Md5值,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端不对移动应用ReactNative模块的配置信息进行热更新。

[0047] 第三方面,本发明提供一种基于ReactNative的多模块移动应用的热更新系统,所述系统包括客户端和服务端;

[0048] 所述客户端,用于进入移动应用的ReactNative模块,加载所述移动应用的jsbundle文件和资源文件,根据所述jsbundle文件和资源文件,向服务器请求所述ReactNative模块的热更新配置信息,所述热更新配置信息包括以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值,接收服务器返回的热更新配置信息,根据所述热更新配置信息进行热更新资源的下载,解压所述热更新资源,将解压后的所述热更新资源缓存到所述移动应用的沙盒路径中;

[0049] 所述服务器,用于接收客户端上传的对移动应用ReactNative模块的配置信息进行热更新的请求,所述热更新的请求中携带以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值,根据所述模块唯一标识,查询服务器的数据库中启用的最新资源,识别是否存在所述最新资源,如果不存在所述最新资源,向客户端返回请求失败信息,如果存在所述最新资源,判断所述请求中携带的应用版本号是否大于或等于服务器的数据库中当前配置的应用版本号,如果所述请求中携带的应用版本号大于或等于服务器的数据库中当前配置的应用版本号,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新。

[0050] 由以上技术方案可知,本发明提供一种基于ReactNative的多模块移动应用的热更新方法及系统,客户端可针对不同ReactNative模块,利用本地的缓存文件或者内置文

件进行热更新,即使网络等原因造成模块资源获取失败,也可以获取缓存中最近的资源文件进行界面数据渲染,服务器端可以根据客户端的请求,精准地为模块下发更新资源,能够更加快速的进行App的版本更新、及时解决线上问题,让用户不用重新下载新版本就能体验新版本App的功能。

附图说明

[0051] 为了更清楚地说明本发明的技术方案,下面将对实施例中所需要使用的附图作简单地介绍,显而易见地,对于本领域普通技术人员而言,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0052] 图1为本发明提供的基于ReactNative的多模块移动应用的热更新方法一实施例的流程图。

[0053] 图2为步骤S102的流程图。

[0054] 图3为步骤S202的流程图。

[0055] 图4为本发明提供的基于ReactNative的多模块移动应用的热更新方法一种可实现方式的流程图。

[0056] 图5为本发明提供的基于ReactNative的多模块移动应用的热更新方法刷新页面的流程图。

[0057] 图6为本发明提供的基于ReactNative的多模块移动应用的热更新方法加载更多数据的流程图。

[0058] 图7为本发明提供的基于ReactNative的多模块移动应用的热更新方法另一实施例的流程图。

[0059] 图8为本发明提供的基于ReactNative的多模块移动应用的热更新系统的示意图。

具体实施方式

[0060] 为使本发明的目的、技术方案和优点更加清楚,下面将结合本发明具体实施例及相应的附图对本发明技术方案进行清楚、完整地描述。显然,所描述的实施例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他实施例,都属于本发明保护的范围。以下结合附图,详细说明本发明各实施例提供的技术方案。

[0061] 请参阅图1,为本发明提供的一种基于ReactNative的多模块移动应用的热更新方法的第一实施例的流程图,该方法可应用于对运行iOS、Android操作系统的客户端上的基于ReactNative多模块的App进行热更新。

[0062] 在本实施例中,所述的热更新就是针对需求或者bug,更新应用代码和图片等资源文件,让用户在不用升级安装新版本的前提下,使用最新版本功能。ReactNative模块会将所有JS代码和图片根据平台,压缩成一个jsbundle文件和一个资源文件。而多模块下的ReactNative应用,则会生成不同模块的对应文件。服务器负责管理这些文件,并且根据客户端请求所传参数,下发对应模块的热更新配置。客户端根据返回的配置信息,下载更新模块最新资源数据,客户端具体可以是智能手机。当客户端需要热更新App某个模块时,则需要考虑模块版本、数据缓存、热更新配置获取、以及最后的数据刷新等。

[0063] 本发明提供了一种基于ReactNative的多模块移动应用的热更新方法,具体可以包括如下步骤:

[0064] 步骤S101,进入移动应用的ReactNative模块。

[0065] 步骤S102,加载所述移动应用的jsbundle文件和资源文件。

[0066] 如图2所示,加载所述移动应用的jsbundle文件和资源文件具体可以包括:

[0067] 步骤S201,检测所述移动应用的沙盒中是否缓存有所述ReactNative模块的jsbundle文件和资源文件。

[0068] 步骤S202,如果所述移动应用的沙盒中缓存有所述ReactNative模块的jsbundle文件和资源文件,加载缓存的jsbundle文件和资源文件。

[0069] 步骤S203,如果所述移动应用的沙盒中没有缓存所述ReactNative模块的jsbundle文件和资源文件,加载所述移动应用中内置的jsbundle文件和资源文件,例如ipa或者apk格式的移动应用中内置的jsbundle文件和资源文件,从而即使网络等原因造成模块资源获取失败,也可以获取缓存中最近的资源文件进行界面数据渲染。

[0070] 如图3所示,进一步地,加载缓存的jsbundle文件和资源文件具体可以包括:

[0071] 步骤S301,将所述ReactNative模块的文件拷贝到currentReadedModule文件中。

[0072] 步骤S302,返回拷贝后的文件路径。

[0073] 步骤S303,读取所述拷贝后的文件路径中的jsbundle文件和资源文件包括进行界面渲染。这样做的目的是避免同时读写操作某模块文件,造成文件错乱、死锁等情况。

[0074] 步骤S103,根据所述jsbundle文件和资源文件,向服务器请求所述ReactNative模块的热更新配置信息,所述热更新配置信息包括以下参数:模块唯一标识(英文:Module Id)、应用版本号(英文:App Version)、jsbundle版本号(英文:Package Version),以及jsbundle的Md5值(英文:Package Md5),服务器根据参数,返回该模块的配置,包含是否需要更新等信息。

[0075] 步骤S104,接收服务器返回的热更新配置信息。

[0076] 步骤S105,根据所述热更新配置信息进行热更新资源的下载。

[0077] 步骤S106,解压所述热更新资源。

[0078] 具体地,如果解压未成功,则可以移除残留的文件。

[0079] 步骤S107,将解压后的所述热更新资源缓存到所述移动应用的沙盒路径中。

[0080] 具体地,在本实施例中,缓存的文件目录结构可以如下:首先,iOS操作系统下为Library目录,Android操作系统下为files目录。Library目录或者files目录下包含app name,即应用名目录。app name目录下包含001、002等目录,模块ID的目录。001、002等目录下分别包含assets,即图片等资源。Jsbundle,即压缩的代码文件。以及config,即资源的配置信息。而Library目录或者files目录下还包含currentReadedModule目录,用于存放正在读取的资源。

[0081] 请参阅图4,在本发明一可选实施方式中,将解压后的所述热更新资源缓存到所述移动应用的沙盒路径中之后,所述方法还可以包括:

[0082] 步骤S401,识别所述ReactNative模块是否需要强制更新。

[0083] 步骤S402,如果所述ReactNative模块需要强制更新,刷新所述ReactNative模块。

[0084] 如图5所示,刷新所述ReactNative模块具体可以包括:

[0085] 步骤S501,向服务器上传刷新页面的请求,所述刷新页面的请求中携带有请求参数,以使得服务器根据所述请求参数查询出未刷新期间产生的数据和补传数据,判断所述未刷新期间产生的数据和补传数据的数据量是否超出预设的最大数据量,在所述未刷新期间产生的数据和补传数据的数据量超出预设的最大数据量时,向客户端返回数据并在返回的数据中携带还有更多数据的标识。

[0086] 具体地,客户端一般为下拉刷新或第一次进入页面时刷新。客户端上传的参数可以包括本地缓存数据中最早的发生时间,以及本地缓存数据中最大唯一有序递增字段值。如果没有缓存则不上传本地缓存数据中最早的发生时间,如果没有缓存的事件则不上传本地缓存数据中最大唯一有序递增字段值。

[0087] 服务器在收到客户端上传的参数之后,可以查询出数据发生时间晚于客户端传入最早发生时间,且唯一有序递增字段值大于客户端传入的本地缓存数据中最大唯一有序递增字段值的相应值,若查询出的这些相应值的数据多于约定的最大数据条数,则服务器返回参数中给予还有更多数据的标识。

[0088] 步骤S502,接收服务器返回的数据。

[0089] 步骤S503,根据服务器返回的数据中的标识,判断是否有更多数据未传回。

[0090] 步骤S504,如果有更多数据未传回,根据服务器返回的数据的最早产生时间,移除本地缓存中所有比所述最早产生时间早的数据。其他数据正常显示。如果没有更多数据未传回,则根据数据的产生时间从晚到早显示在界面上。

[0091] 步骤S505,将其余的数据缓存于本地数据库中。

[0092] 步骤S506,刷新显示界面。

[0093] 进一步地,请参阅图6,所述方法还可以包括加载更多数据时的步骤:

[0094] 步骤S601,向服务器上传加载更多数据的请求,所述加载更多数据的请求中携带有客户端当前界面显示数据中的最早发生时间,以使得服务器查找出比所述最早发生时间更早的数据,将比所述最早发生时间更早的数据返回至客户端。

[0095] 具体地,客户端一般为上拉加载更多数据。

[0096] 步骤S602,接收服务器返回的比所述最早发生时间更早的数据。

[0097] 步骤S603,将比所述最早发生时间更早的数据缓存本地数据库中。

[0098] 步骤S604,刷新显示界面。

[0099] 此外,本发明可实现一套可视化的前端操作界面(H5页面),可以添加资源、参数的可选配置。根据资源上传接口,实现不同应用不同模块的资源上传管理。其中,可配置的参数如下:资源类型(对应模块id)、名称(应用版本号)、描述(jsbundle版本号)、版本、最低适配的应用版本、平台(iOS/Android)、是否强制更新、资源包(zip文件)以及资源是否启用。可视化的操作界面,简化了热更新资源的上传流程。测试、运营都可根据需求,填写配置参数,进行文件的上传。同时,也可配置用户权限,方便对热更新文件进行操作管理。

[0100] 请参阅图7,为一种基于ReactNative的多模块移动应用的热更新方法,应用于服务器,所述方法可以包括如下步骤:

[0101] 步骤S701,接收客户端上传的对移动应用ReactNative模块的配置信息进行热更新的请求,所述热更新的请求中携带以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值。

- [0102] 步骤S702,根据所述模块唯一标识,查询服务器的数据库中启用的最新资源。
- [0103] 步骤S703,识别是否存在所述最新资源。如果不存在所述最新资源,向客户端返回请求失败信息。
- [0104] 如果存在所述最新资源,则执行步骤S704,判断所述请求中携带的应用版本号是否大于或等于服务器的数据库中当前配置的应用版本号。
- [0105] 步骤S704,如果所述请求中携带的应用版本号大于或等于服务器的数据库中当前配置的应用版本号,执行步骤S707,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新。
- [0106] 如果所述请求中携带的应用版本号不大于或等于服务器的数据库中当前配置的应用版本号,则执行执行步骤S705,判断所述请求中携带的jsbundle版本号是否大于或等于服务器的数据库中当前配置的jsbundle版本号。
- [0107] 如果所述请求中携带的jsbundle版本号大于或等于服务器的数据库中当前配置的jsbundle版本号,执行步骤S707,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新。
- [0108] 如果所述请求中携带的jsbundle版本号不大于或等于服务器的数据库中当前配置的jsbundle版本号,则执行执行步骤S706,判断所述请求中携带的jsbundle的Md5值是否大于或等于服务器的数据库中当前配置的jsbundle的Md5值。
- [0109] 如果所述请求中携带的jsbundle的Md5值大于或等于数据库中当前配置的jsbundle的Md5值,执行步骤S707,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新。
- [0110] 如果所述请求中携带的jsbundle的Md5值不大于或等于服务器的数据库中当前配置的jsbundle的Md5值,则执行步骤S708,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端不对移动应用ReactNative模块的配置信息进行热更新。
- [0111] 请参阅图8,本发明还提供一种基于ReactNative的多模块移动应用的热更新系统,所述系统包括客户端81和服务器82。
- [0112] 客户端81用于进入移动应用的ReactNative模块,加载移动应用的jsbundle文件和资源文件,根据jsbundle文件和资源文件,向服务器请求ReactNative模块的热更新配置信息,热更新配置信息包括以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值,接收服务器返回的热更新配置信息,根据热更新配置信息进行热更新资源的下载,解压热更新资源,将解压后的热更新资源缓存到移动应用的沙盒路径中。
- [0113] 服务器82用于接收客户端上传的对移动应用ReactNative模块的配置信息进行热更新的请求,热更新的请求中携带以下参数:模块唯一标识、应用版本号、jsbundle版本号,以及jsbundle的Md5值,根据模块唯一标识,查询服务器的数据库中启用的最新资源,识别是否存在最新资源,如果不存在最新资源,向客户端返回请求失败信息,如果存在最新资源,判断请求中携带的应用版本号是否大于或等于服务器的数据库中当前配置的应用版本

号,如果请求中携带的应用版本号大于或等于服务器的数据库中当前配置的应用版本号,向客户端返回服务器的数据库中当前配置的移动应用ReactNative模块的配置信息,以使客户端对移动应用ReactNative模块的配置信息进行热更新。

[0114] 综上所述,本发明实施例提供一种基于ReactNative多模块下的热更新方法及系统,客户端可针对不同ReactNative模块,利用本地的缓存文件或者内置文件进行热更新,即使网络等原因造成模块资源获取失败,也可以获取缓存中最近的资源文件进行界面数据渲染,服务器端可以根据客户端的请求,精准地为模块下发更新资源,能够更加快速的进行App的版本更新、及时解决线上问题,让用户不用重新下载新版本就能体验新版本App的功能。并且,模块的热更新规则由服务器维护,可根据需求,实时更改规则。还实现了可视化的上传界面,方便了资源上传的流程,便于管理者操作热更新文件。

[0115] 本发明实施例还提供一种存储介质,本发明实施例还提供一种存储介质,所述存储介质中存储有计算机程序,所述计算机程序被处理器执行时实现本发明提供的基于ReactNative多模块下的热更新方法的各实施例中的部分或全部步骤。所述的存储介质可为磁碟、光盘、只读存储记忆体(英文:Read-OnlyMemory,简称:ROM)或随机存储记忆体(英文:RandomAccessMemory,简称:RAM)等。

[0116] 本领域的技术人员可以清楚地了解到本发明实施例中的技术可借助软件加必需的通用硬件平台的方式来实现。基于这样的理解,本发明实施例中的技术方案本质上或者说对现有技术做出贡献的部分可以以软件产品的形式体现出来,该计算机软件产品可以存储在存储介质中,如ROM/RAM、磁碟、光盘等,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,服务器,或者网络设备)执行本发明各个实施例或者实施例的某些部分所述的方法。

[0117] 本说明书中各个实施例之间相同相似的部分互相参见即可。尤其,对于基于ReactNative多模块下的热更新系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例中的说明即可。

[0118] 以上所述的本发明实施方式并不构成对本发明保护范围的限定。



图1

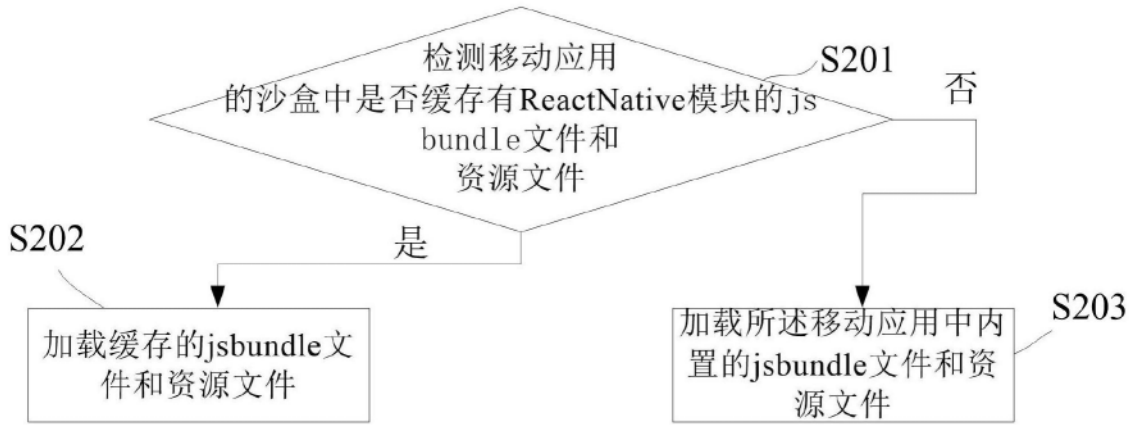


图2

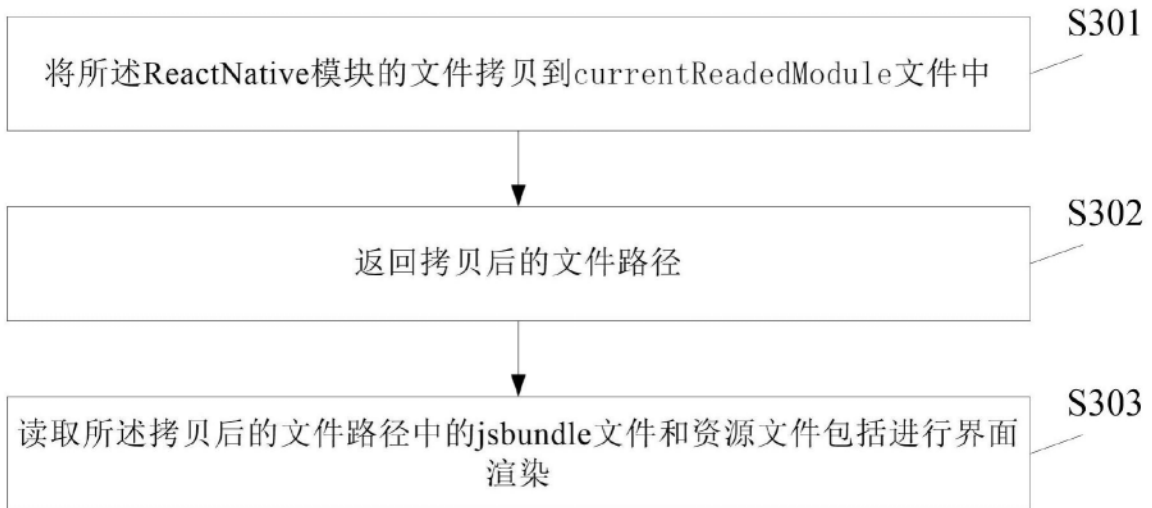


图3

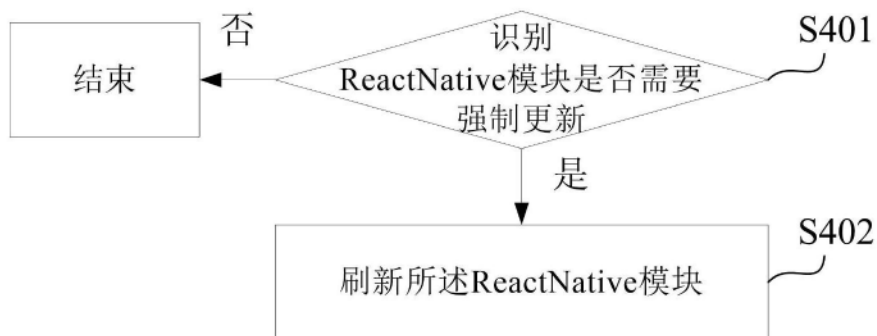


图4

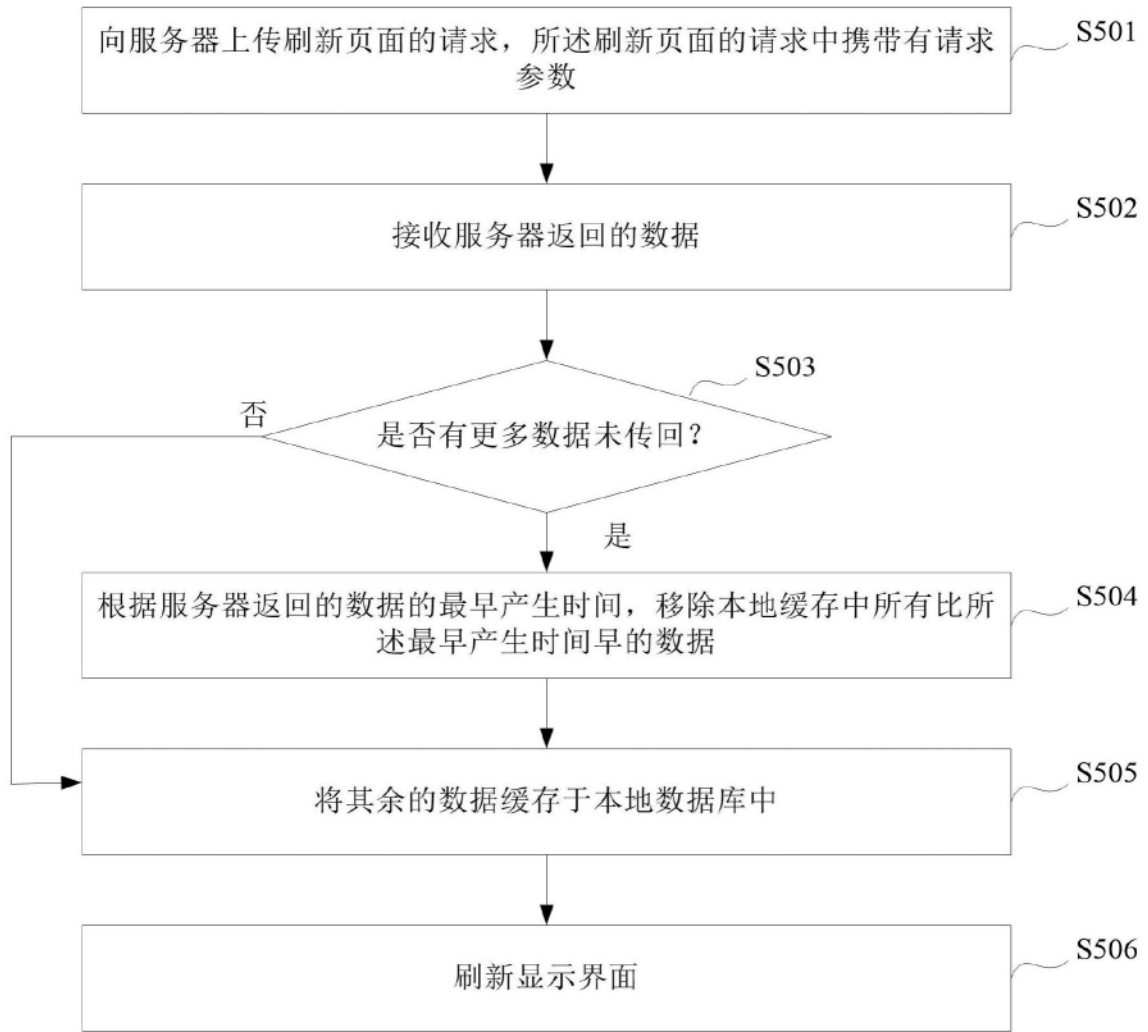


图5

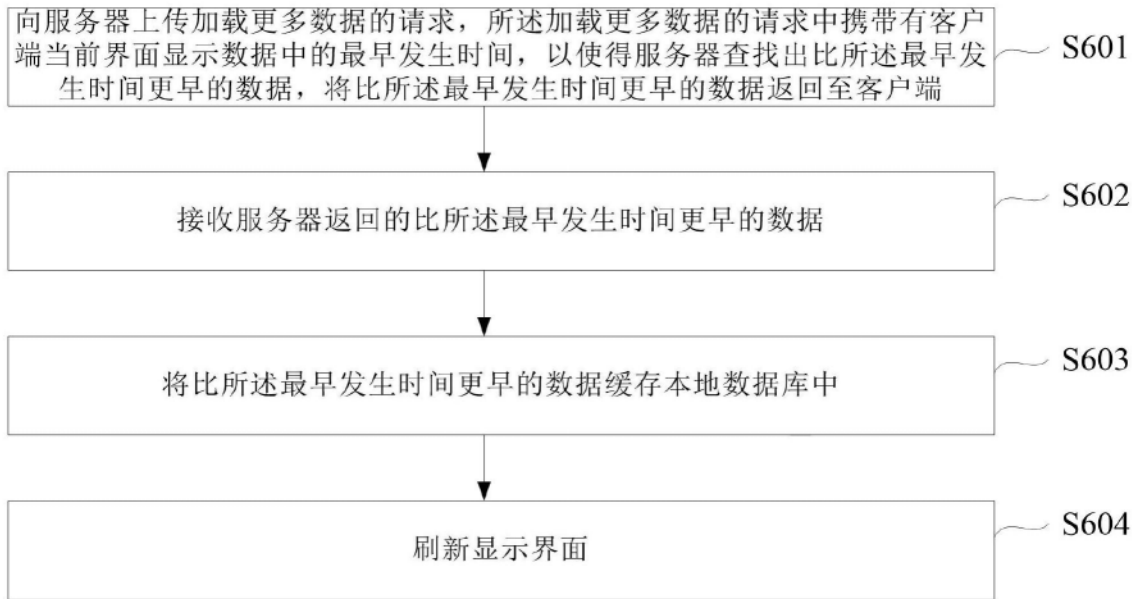


图6

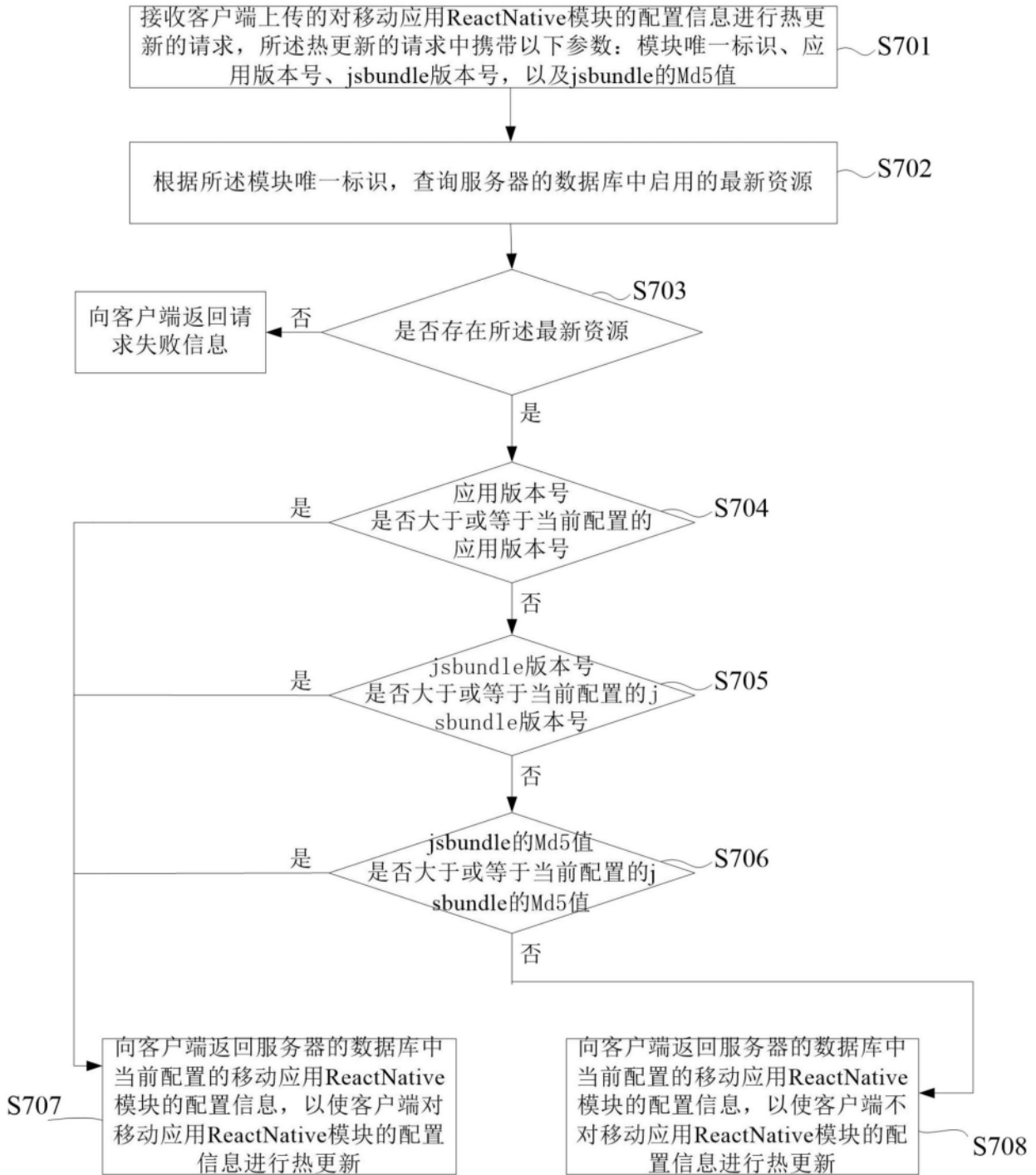


图7

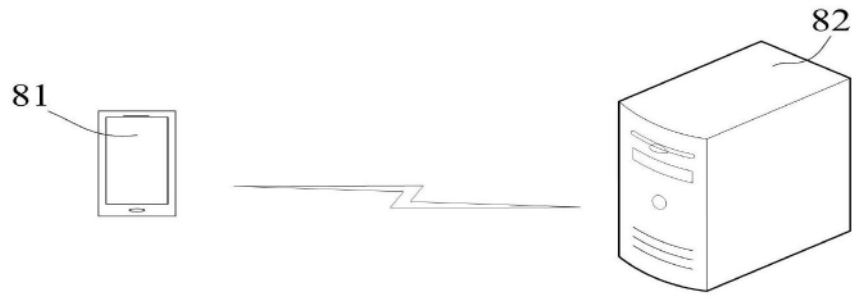


图8