

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2020 年 2 月 6 日 (06.02.2020)



(10) 国际公布号
WO 2020/024399 A1

- (51) 国际专利分类号:
G06F 16/957 (2019.01)
- (21) 国际申请号: PCT/CN2018/106434
- (22) 国际申请日: 2018 年 9 月 19 日 (19.09.2018)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
201810870260.2 2018年8月2日 (02.08.2018) CN
- (71) 申请人: 平安科技(深圳)有限公司(PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) [CN/CN];
中国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, Guangdong 518000 (CN)。
- (72) 发明人: 徐唐生(XU, Tangsheng); 中国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, Guangdong 518000 (CN)。
- (74) 代理人: 深圳众鼎专利商标代理事务所(普通合伙)(SHENZHEN ZHONGDING INTELLECTUAL PROPERTY AGENCY); 中国广东省深圳市龙岗区龙城街道中心城清林路 546 号城投商务中心 4 层/B, Guangdong 518172 (CN)。
- (81) 指定国(除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL,

(54) Title: PAGE RENDERING METHOD AND APPARATUS, COMPUTER DEVICE, AND STORAGE MEDIUM

(54) 发明名称: 页面渲染方法、装置、计算机设备及存储介质

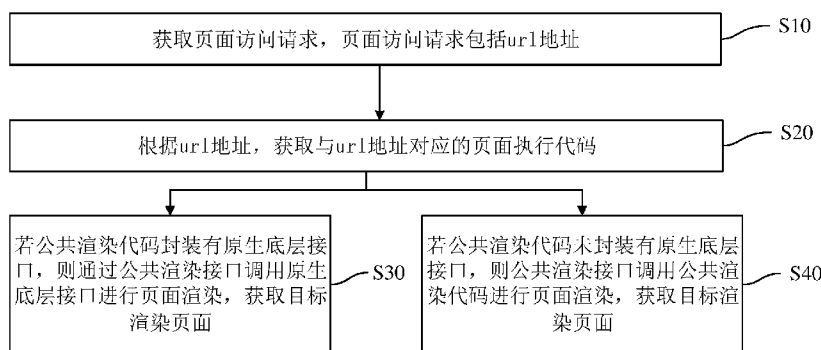


图 2

- S10 Acquire a page access request, wherein the page access request comprises a URL address
- S20 According to the URL address, acquire a page execution code corresponding to the URL address
- S30 If a common rendering code is encapsulated with a native underlying interface, invoke the native underlying interface by means of a common rendering interface in order to carry out page rendering so as to acquire a target rendered page
- S40 If a common rendering code is not encapsulated with a native underlying interface, invoke the common rendering code by means of a common rendering interface in order to carry out page rendering so as to acquire a target rendered page

(57) Abstract: A page rendering method and apparatus, a computer device, and a storage medium. The method comprises: acquiring a page access request, wherein the page access request comprises a URL address (S10); according to the URL address, acquiring a page execution code corresponding to the URL address (S20), wherein the page execution code comprises at least one common rendering interface, and each common rendering interface corresponds to a common rendering code; if the common rendering code is encapsulated with a native underlying interface, invoking the native underlying interface by means of the common rendering interface in order to carry out page rendering so as to acquire a target rendered page (S30); and if a common rendering code is not encapsulated with a

PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL,
SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG,
US, UZ, VC, VN, ZA, ZM, ZW。

- (84) 指定国(除另有指明, 要求每一种可提供的地区
保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ,
NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM,
AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG,
CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU,
IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT,
RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI,
CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

- 包括国际检索报告(条约第21条(3))。

native underlying interface, invoking a common rendering code by means of the common rendering interface in order to carry out page rendering so as to acquire a target rendered page (S40). The problems of jumping between pages not being smooth and blanks appearing on the pages are solved.

(57) 摘要: 一种页面渲染方法、装置、计算机设备及存储介质, 该方法包括: 获取页面访问请求, 页面访问请求包括url地址(S10); 根据url地址, 获取与url地址对应的页面执行代码(S20), 页面执行代码包括至少一个公共渲染接口, 每一公共渲染接口对应一公共渲染代码; 若公共渲染代码封装有原生底层接口, 则通过公共渲染接口调用原生底层接口进行页面渲染, 获取目标渲染页面(S30); 若公共渲染代码未封装有原生底层接口, 则公共渲染接口调用公共渲染代码进行页面渲染, 获取目标渲染页面(S40)。解决页面之间的跳转不流畅, 页面出现空白的问题。

页面渲染方法、装置、计算机设备及存储介质

本申请以 2018 年 8 月 2 日提交的申请号为 201810870260.2，名称为“页面渲染方法、装置、计算机设备及存储介质”的中国发明申请为基础，并要求其优先权。

技术领域

本申请涉及计算机领域领域，尤其涉及一种页面渲染方法、装置、计算机设备及存储介质。

背景技术

目前市面上的 H5 型 APP 中，大多是采用 H5 对页面进行渲染，但是通过 H5 进行页面渲染，会比较消耗性能，从而导致页面之间的跳转不流畅，使得页面出现空白的问题。

发明内容

本申请实施例提供一种页面渲染方法、装置、计算机设备及存储介质，以解决采用 H5 对页面进行渲染而导致页面之间的跳转不流畅的问题。

一种页面渲染方法，包括：

获取页面访问请求，所述页面访问请求包括 url 地址；

根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，所述页面执行代码包括至少一个公共渲染接口，每一所述公共渲染接口对应一公共渲染代码；

若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面；

若所述公共渲染代码未封装有原生底层接口，则所述公共渲染接口调用所述公共渲染代码进行页面渲染，获取目标渲染页面。

一种页面渲染装置，包括：

请求获取模块，用于获取页面访问请求，所述页面访问请求包括 url 地址；

代码获取模块，用于根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，所述页面执行代码包括至少一个公共渲染接口，每一所述公共渲染接口对应一公共渲染代码；

底层接口渲染模块，用于若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面；

代码渲染模块，用于若所述公共渲染代码未封装有原生底层接口，则所述公共渲染接口调用所述公共渲染代码进行页面渲染，获取目标渲染页面。

一种计算机设备，包括存储器、处理器以及存储在存储器中并可在处理器上运行的计算机可读指令，处理器执行计算机可读指令时实现如下步骤：

获取页面访问请求，所述页面访问请求包括 url 地址；

根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，所述页面执行代码包括

至少一个公共渲染接口，每一所述公共渲染接口对应一公共渲染代码；

若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面；

若所述公共渲染代码未封装有原生底层接口，则所述公共渲染接口调用所述公共渲染代码进行页面渲染，获取目标渲染页面。

一个或多个存储有计算机可读指令的非易失性可读存储介质，计算机可读存储介质存储有计算机可读指令，使得所述一个或多个处理器执行如下步骤，所述计算机可读指令被处理器执行时实现如下步骤：

获取页面访问请求，所述页面访问请求包括 url 地址；

根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，所述页面执行代码包括至少一个公共渲染接口，每一所述公共渲染接口对应一公共渲染代码；

若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面；

若所述公共渲染代码未封装有原生底层接口，则所述公共渲染接口调用所述公共渲染代码进行页面渲染，获取目标渲染页面。

本申请的一个或多个实施例的细节在下面的附图及描述中提出。本申请的其他特征和优点将从说明书、附图以及权利要求书变得明显。

附图说明

为了更清楚地说明本申请实施例的技术方案，下面将对本申请实施例的描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本申请的一些实施例，对于本领域普通技术人员来讲，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。

图 1 是本申请一实施例中页面渲染方法的一应用环境示意图；

图 2 是本申请一实施例中页面渲染方法的一流程图；

图 3 是本申请一实施例中页面渲染方法的一流程图；

图 4 是本申请一实施例中页面渲染方法的一流程图；

图 5 是本申请一实施例中页面渲染方法的一流程图；

图 6 是本申请一实施例中页面渲染方法的一流程图；

图 7 是本申请一实施例中页面渲染装置的一示意图；

图 8 是本申请一实施例中计算机设备的一示意图。

具体实施方式

下面将结合本申请实施例中的附图，对本申请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申请一部分实施例，而不是全部的实施例。基于本实施例中的实施例，本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例，都属于本申请保护的范围。

本实施例中提供的页面渲染方法，可应用在如图 1 的应用环境中，其中，用户端通过网络与服务器进行通信。其中，用户端可以但不限于各种个人计算机、笔记本电脑、智能手机、平板电脑和便携式可穿戴设备。服务器可以用独立的服务器或者是多个服务器组成的服务器集群来实现。

在一实施例中，如图 2 所示，提供一种页面渲染方法，以该方法应用在图 1 中的服务器为例进行说明，具体包括如下步骤：

S10: 获取页面访问请求，页面访问请求包括 url 地址。

其中，页面访问请求是指用户端向服务器发送的访问页面的请求。例如，访问 APP 中某一页面时，APP 提供一可视化页面，用户端可通过在该可视化页面进行相关操作（如点击或滑动等）向服务器发送需要进入与某一页面相对应的页面访问请求，以使服务器可获取到该页面访问请求。

具体地，用户端安装的 APP 为混合 H5 型 APP，混合 H5 型 APP 中可通过原生底层接口和 H5 对页面进行渲染。具体地，混合 H5 型 APP 中引入基于 JS 开发的前端框架 mui 作为开发框架，将前端工具封装成开发的 JavaScript 方法，为通过调用原生底层接口和 H5 对页面进行渲染提供技术基础。用户端通过网络与服务器进行通信连接，当服务器接收用户端基于混合 H5 型 APP 触发并发送的页面访问请求时，获取页面访问请求中 url 地址。可以理解地，服务器可通过域名解析器解析出 url 地址中的 ip 地址，以便对该页面访问请求进行响应。

其中，H5（HTML5 的简称）是集文字、图片、音乐、视频和链接等多种形式的展示页面，具有丰富的控件、灵活的动画特效、强大的交互应用和数据分析以及高速低价的实现信息传播等优点，非常适合通过用户端的展示和分享。

其中，JavaScript(简称 JS)是一种直译式脚本语言，是一种动态类型、弱类型和基于原型的语言，内置支持类型。它的解释器被称为 JavaScript 引擎，为浏览器的一部分，广泛用于用户端的脚本语言，最早是在 HTML(标准通用标记语言下的一个应用)网页上使用，用来给 HTML 网页增加动态功能。

其中，url 地址是(Uniform Resource Locator，统一资源定位符)的缩写，它是 WWW 的统一资源定位标志，即指网络地址。

S20: 根据 url 地址，获取与 url 地址对应的页面执行代码。

其中，页面执行代码是指与 url 地址对应的网页源代码，本实施例中，可以是 JavaScript 代码。当服务器获取到用户端发送的页面访问请求时，获取页面访问请求中的 url 地址，根据 url 地址获取与 url 地址对应的页面执行代码，以便后续通过页面执行代码进行页面渲染。可以理解地，服务器在获取 url 地址对应的页面执行代码之后，可基于该页面执行代码进行页面渲染，获取目标渲染页面。

进一步地，步骤 S20 中获取到的页面执行代码包括至少一个公共渲染接口，每一公共渲染接口对应一公共渲染代码。

其中，公共渲染接口是指对页面进行公共处理的接口，公共渲染接口中包括原生底层接口，还可以包括公共渲染代码，通过公共渲染接口执行原生底层接口或公共渲染代码进行页面渲染。公共渲染代码是指对页面进行公共渲染的代码，公共渲染代码是一些预先定义的渲染函数，公共渲染代码为用户提供基于某软件或硬件得以访问一组例程的能力，而又无需访问源码或理解内部工作机制的细节。

具体地，服务器获取 url 地址，根据 url 地址获取对应的页面执行代码，即 JavaScript 代码，JavaScript 代码中包括至少一个公共渲染接口，公共渲染接口中对公共渲染代码进行封装，当执行到公共渲染接口时，则通过公共渲染接口中公共渲染代码对页面进行渲染，其中，每一公共渲染接口对应一功能模块。其中，功能模块是指对页面中不同功能进行渲染的模块，如页面中包含顶部导航和左右滑动等功能模块。每一页面中包括至少一个功能模块，每一功能模块对应一公共渲染接口，如一功能模块为顶部导航，则其对应的公共渲染接口为实现该顶部导航的公共渲染接口。

S30: 若公共渲染代码封装有原生底层接口，则通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面。

其中，原生底层接口是指通过公共渲染代码来调用相关的接口从而间接调用底层驱动功能的接口。原生底层接口具有访问操作系统的预先定义的渲染函数的功能。

服务器通过页面执行代码对页面进行渲染，该页面执行代码包括至少一个公共渲染接口，当公共渲染接口中的公共渲染代码中封装有原生底层接口时，则通过公共渲染接口调用原生底层接口进行页面渲染，即通过公共渲染代码调用与原生底层接口相对应的渲染函数进行页面渲染，获取目标渲染页面。例如，一公共渲染接口中封装有可实现对页面的标题和箭头进行渲染的原生底层接口，服务器可通过该公共渲染接口调用原生底层接口进行页面渲染，以实现对该页面的标题和箭头进行渲染，获取目标渲染页面。本实施例中，服务器通过原生底层接口进行页面渲染，无需关注和处理复杂的网络协议内容，立即对原生底层接口渲染的目标渲染页面进行展示，提高性能，解决页面出现空白的问题，在无网络或网络不好的情况下，渲染速度快，提高页面跳转的流畅性。

S40: 若公共渲染代码未封装有原生底层接口，则公共渲染接口调用公共渲染代码进行页面渲染，获取目标渲染页面。

服务器通过页面执行代码对页面进行渲染，该页面执行代码包括至少一个公共渲染接口，若公共渲染接口中只封装有公共渲染代码而未封装有原生底层接口时，则通过公共渲染接口调用公共渲染代码进行页面渲染，获取目标渲染页面。通过公共渲染接口调用公共渲染代码进行页面渲染，可完成复杂的图标处理和渲染，以实现对该页面进行公共处理，减少渲染时间。

步骤 S10-S40 中，通过获取页面访问请求，页面访问请求包括 url 地址，根据 url 地址，获取与 url 地址对应的页面执行代码，以便后续根据页面执行代码进行页面渲染。页面执行代码包括至少一个公共渲染接口，每一公共渲染接口对应一公共渲染代码，该公共渲染代码可实现对该公共渲染接口对应的功能模块的页面渲染，以使基于页面执行代码渲染出的目标渲染页面上呈现多个功能模块的渲染效果。当公共渲染代码封装有原生底层接口，则通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面，通过原生底层

接口进行页面渲染，并立即对原生底层接口渲染的页面进行展示，提高性能，解决页面出现空白的问题。若公共渲染代码未封装有原生底层接口，则公共渲染接口调用公共渲染代码进行页面渲染，获取目标渲染页面，以实现对该页面进行公共处理，提高性能，减少渲染时间。

在一实施例中，页面包括主页面和内容页面，页面执行代码包括主页面执行代码和内容页面执行代码，通过主页面执行代码对页面的主页面进行渲染，通过内容页面执行代码对页面的内容页面进行渲染。

如图 3 所示，在步骤 S20 中，即根据 url 地址，获取与 url 地址对应的页面执行代码，具体包括如下步骤：

S21：判断页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址。

其中，当前页面的 url 地址是指在 APP 中当前显示的页面的 url 地址。页面访问请求的 url 地址是指本次页面访问请求所要访问的页面的 url 地址，即当前页面的下一访问页面的 url 地址。ip 地址（Internet Protocol Address，简称 ip 地址）是一种在 Internet 上的给主机编址的方式，也称为网际协议地址。ip 地址是 ip 协议提供的一种统一的地址格式，它为互联网上的每一个网络和每一台主机分配一个逻辑地址，以此来屏蔽物理地址的差异。

具体地，服务器获取用户端发送的页面访问请求，页面访问请求中包括 url 地址，提取页面访问请求对应的 url 地址和当前页面对应的 url 地址，并通过域名解析器解析出 url 地址中的 ip 地址，判断与页面访问请求对应的 ip 地址和当前页面对应的 ip 地址是否为同一 ip 地址；若与页面访问请求对应的 ip 地址和当前页面对应的 ip 地址相同，则对应同一 ip 地址；若与页面访问请求对应的 ip 地址和当前页面对应的 ip 地址不相同，则不对应同一 ip 地址。

S22：若对应同一 ip 地址，则根据 url 地址，获取与 url 地址对应的内容页面执行代码。

具体地，当服务器判断出与页面访问请求对应的 ip 地址与当前页面对应的 ip 地址相同时，根据与页面访问请求对应的 url 地址，获取与 url 地址对应的内容页面执行代码，以便后续通过该内容页面执行代码对内容页面进行渲染。由于当与页面访问请求对应的 ip 地址和当前页面对应的 ip 地址相同时，该页面访问请求对应的主页面不需进行页面渲染，可直接显示当前页面的主页面，因此只需根据 url 地址，获取与 url 地址对应的内容页面执行代码。

本实施例中，基于该内容页面执行代码进行页面渲染，获取基于内容页面执行代码渲染出的内容页面，并将该内容页面填充至当前页面的主页面中，形成目标渲染页面，以节省对主页面进行渲染的时间，直接显示主页面，提高用户体验，避免页面出现空白的问题。

S23：若不对应同一 ip 地址，则根据 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执行代码。

具体地，当服务器判断出与页面访问请求对应的 ip 地址和当前页面对应的 ip 地址不相同，认定两者不对应同一 ip 地址，即该页面访问请求对应的主页面和内容页面都需进行页面渲染，因此，服务器需根据与页面访问请求对应的 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执行代码，以便后续通过主页面执行代码和内容页面执行代码对页面的主页面和内容页面进行渲染。

步骤 S21-S23 中, 通过判断页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址, 若对应同一 ip 地址, 则根据 url 地址, 获取与 url 地址对应的内容页面执行代码, 无需获取主页面执行代码, 以便后续页面渲染过程中节省对主页面进行渲染的时间, 避免页面出现空白的问题; 若不对应同一 ip 地址, 则根据 url 地址, 获取与 url 地址对应的主页面执行代码和内容页面执行代码, 以便后续基于该主页面执行代码和内容页面执行代码进行页面渲染, 以实现通过主页面执行代码和内容页面执行代码对页面进行渲染, 减少渲染时间, 提高页面跳转的流畅性。

在一实施例中, 服务器基于页面执行代码进行页面渲染, 获取目标渲染页面, 页面执行代码中包括对目标渲染页面进行渲染的代码, 通过页面执行代码进行页面渲染, 获取目标渲染页面。由于 url 地址对应的页面包括主页面和内容页面, 则该页面执行代码包括对主页面进行渲染的主页面执行代码和对内容页面进行渲染的内容页面执行代码, 以便后续基于该主页面执行代码和内容页面执行代码分别对主页面和内容页面进行渲染, 以获取目标渲染页面。

本实施例中, 如图 4 所示, 在步骤 S23 之后, 即在根据 url 地址, 获取与 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后, 页面渲染方法还包括如下步骤:

S2311: 根据并行处理方式分别对主页面执行代码和内容页面执行代码进行页面渲染, 分别获取主页面和内容页面。

具体地, 当服务器根据 url 地址, 获取与 url 地址对应的主页面执行代码和内容页面执行代码后, 通过并行处理方式分别通过主页面执行代码和内容页面执行代码进行页面渲染, 即通过主页面执行代码进行页面渲染, 以获取主页面; 并通过内容页面执行代码进行页面渲染, 以获取内容页面。通过将主页面执行代码和内容页面执行代码分配到不同处理进程或线程中, 并行对同一页面的不同执行代码进行页面渲染, 以提高执行速度, 减少渲染时间, 提高用户体验。

进一步地, 当服务器根据 url 地址, 获取与 url 地址对应的主页面执行代码和内容页面执行代码, 其中, 主页面执行代码和内容页面执行代码中可包括原生底层接口和公共渲染代码, 并通过原生底层接口或公共渲染代码进行页面渲染。通过主页面执行代码和内容页面执行代码中包括原生底层接口和公共渲染代码, 以便后续通过原生底层接口和公共渲染代码进行页面渲染, 减少渲染时间。

S2312: 根据叠加方式对主页面和内容页面进行叠加, 获取目标渲染页面。

具体地, 服务器获取主页面和内容页面, 通过叠加方式对主页面和内容页面进行叠加, 获取目标渲染页面。其中, 预先设定有与主页面和内容页面对应的的主页面执行代码和内容页面执行代码, 其中, 主页面执行代码中主要是对标题、箭头、顶部导航、底部选项卡和页面底部颜色等进行设定。内容页面中主要是对页面的具体内容进行设定, 即内容页面执行代码中主要是对页面的文字内容和图片内容等进行设定。本实施例中, 采用叠加方式对主页面和内容进行叠加具体是指通过将内容页面嵌入至主页面进行叠加, 以获取目标渲染页面。

步骤 S2311-S2312 中, 根据并行处理方式分别对主页面执行代码和内容页面执行代码进行页面渲染, 分别获取主页面和内容页面, 以实现同时对同一页面的主页面和内容页面

进行渲染,可加快目标渲染页面的渲染效率。根据叠加方式对主页面和内容页面进行叠加,以快速获取目标渲染页面。服务器通过对主页面和内容页面进行叠加处理,分别对主页面和内容页面进行处理,以实现主页面和内容页面分别渲染,不需全部渲染完成,即对目标渲染页面进行展示,避免页面出现空白。

在一实施例中,如图 5 所示,在步骤 S23 之后,即在根据 url 地址,获取与 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后,页面渲染方法具体还包括如下步骤:

S2321: 判断 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口。

具体地,服务器获取 url 地址对应的主页面执行代码中的公共渲染接口和当前页面的页面执行代码中的公共渲染接口,将 url 地址对应的主页面执行代码中的公共渲染接口与当前页面的页面执行代码中的公共渲染接口进行对比。若 url 地址对应的主页面执行代码中的任一公共渲染接口与当前页面的页面执行代码中的公共渲染接口相同,则判定存在一致的公共渲染接口。若 url 地址对应的主页面执行代码中的所有公共渲染接口与当前页面的页面执行代码中的公共渲染接口均不相同,则判定不存在一致的公共渲染接口。

S2322: 若存在一致的公共渲染接口,则跳过公共渲染接口。

具体地,当服务器判断出 url 地址对应的主页面执行代码中的公共渲染接口与当前页面的页面执行代码中的公共渲染接口相同时,则跳过公共渲染接口,不对公共渲染接口对应的功能模块进行页面渲染。例如,当前页面中包括公共渲染接口 1,若 url 地址对应的主页面执行代码中的公共渲染接口包括公共渲染接口 1,则通过 url 地址对应的主页面执行代码进行页面渲染时,跳过该公共渲染接口。再例如,混合 H5 型 APP 中,当前页面的顶部导航和页面底部颜色由公共渲染接口 1 进行渲染,当 url 地址对应的主页面执行代码中的公共渲染接口包括公共渲染接口 1,则跳过顶部导航和页面底部颜色的渲染,节省相关功能模块的渲染时间。

S2323: 若不存在一致的公共渲染接口,则基于公共渲染接口进行页面渲染,获取目标渲染页面。

具体地,当服务器判断出 url 地址对应的主页面执行代码中的公共渲染接口与当前页面的页面执行代码中的公共渲染接口不相同时,则基于公共渲染接口进行页面渲染,获取目标渲染页面。

步骤 S2321-S2323 中,通过判断 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口;若存在一致的公共渲染接口,则跳过公共渲染接口,节省对一致的公共渲染接口进行渲染的时间;若不存在一致的公共渲染接口,则基于公共渲染接口进行页面渲染,以实现通过判断 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口,确定需要对页面进行渲染的公共渲染接口,通过公共渲染接口进行页面渲染,获取目标渲染页面。

在一实施例中,如图 6 所示,若公共渲染代码封装有原生底层接口,则通过公共渲染接口调用原生底层接口进行页面渲染,获取目标渲染页面的步骤之前,页面渲染方法还包

括如下步骤:

S201: 获取页面渲染配置请求, 页面渲染配置请求包括至少一个样式渲染标识和渲染顺序。

具体地, 服务器获取用户端发送的页面渲染配置请求, 页面渲染配置请求中包括至少一个样式渲染标识和样式渲染标识之间的渲染顺序。其中, 页面渲染配置请求可以是对主页面进行渲染的页面渲染配置请求, 也可以是对内容页面进行渲染的页面渲染配置请求。其中, 样式渲染标识是指对页面中样式进行渲染的标识。样式渲染标识可以是与页面的背景颜色、宽、高和位置等样式对应的标识。例如, 样式渲染标识为 **background**、**top**、**left**、**right**、**bottom** 和 **height** 等。渲染顺序是指对页面中各样式渲染标识对应的样式之间进行渲染的顺序, 例如, 先对页面的长宽高进行渲染, 再对页面的标题、进度条、动画效果和箭头等进行渲染的顺序。

S202: 基于样式渲染标识查询渲染函数库, 判断渲染函数库中是否包含与样式渲染标识对应的渲染函数, 渲染函数包括至少一个形式参数。

其中, 函数渲染库是指存储渲染函数的数据库, 函数渲染库存储有大量的渲染函数, 每一渲染函数对应一样式渲染标识, 以便基于该样式渲染标识查询到对应的渲染函数, 每一渲染函数中包含至少一个形式参数, 在具体应用中, 可将与形式参数对应的实际参数传递给该渲染函数, 以实现该渲染函数的更新。服务器将样式渲染标识与函数渲染库进行匹配处理, 若在渲染函数库中匹配到该样式渲染标识对应的渲染函数, 则判定为渲染函数库中包含与样式渲染标识对应的渲染函数; 若在渲染函数库中未匹配到该样式渲染标识对应的渲染函数, 则判定为渲染函数库中不包含与样式渲染标识对应的渲染函数。例如, 样式渲染标识为 **background**, 通过样式渲染标识查找函数渲染库, 判断渲染函数库中是否包含与 **background** 相对应的渲染函数。再例如, 样式渲染标识为 **height**, 判断渲染函数库中是否包含与 **height** 相对应的渲染函数。

S203: 若渲染函数库中包含与样式渲染标识对应的渲染函数, 则获取与渲染函数中的形式参数相对应的实际参数, 生成与样式渲染标识对应的原生底层接口。

具体地, 服务器在渲染函数库中匹配到与样式渲染标识对应的渲染函数, 获取渲染函数, 将样式渲染标识对应的实际参数赋值给渲染函数的形式参数, 并生成与样式渲染标识对应的原生底层接口。例如, 样式渲染标识为 **background**, 通过样式渲染标识查找函数渲染库, 当渲染函数库中包含与 **background** 相对应的渲染函数, 则可通过渲染函数对页面空白区域的背景色进行设定, 将与 **background** 相对应的实际参数 (即背景色) 赋值给渲染函数的形式参数, 并生成一个原生底层接口, 当调用该原生底层接口时, 可对页面的空白区域的背景色进行渲染。再例如, 样式渲染标识为 **height**, 通过样式渲染标识查找函数渲染库, 当渲染函数库中包含与 **height** 相对应的渲染函数, 则可通过渲染函数对页面高度进行设定, 将与 **height** 相对应的实际参数 (即页面高度) 赋值给渲染函数的形式参数, 并生成一个原生底层接口, 当调用该原生底层接口时, 可对页面的高度进行渲染。

S204: 若渲染函数库中不包含与样式渲染标识对应的渲染函数, 则获取用户配置的与样式渲染标识对应的公共渲染代码。

具体地, 服务器在渲染函数库中未匹配到与样式渲染标识对应的渲染函数, 获取用户

端发送的与样式渲染标识对应的公共渲染代码，通过执行该公共渲染代码，可进行页面渲染。例如，样式渲染标识为 `titleview`，在渲染函数库中未匹配到渲染标识 `titleview`，则判定为渲染函数库中不包含与样式渲染标识 `titleview` 对应的渲染函数，获取用户端发送的与样式渲染标识 `titleview` 对应公共渲染代码。

进一步地，可将配置的与样式渲染标识对应的公共渲染代码保存至渲染函数库中，以便后续若页面渲染配置请求中样式渲染标识与公共渲染代码相对应，则获取与公共渲染代码中的形式参数相对应的实际参数，生成与样式渲染标识对应的原生底层接口，以提高对公共渲染代码的利用率。

S205: 根据渲染顺序，对原生底层接口和公共渲染代码进行封装，获取公共渲染接口。

具体地，服务器根据渲染顺序对原生底层接口和公共渲染代码进行封装，获取公共渲染接口，以便后续通过执行公共渲染接口，根据渲染顺序对页面进行渲染。

步骤 S201-S205 中，获取页面渲染配置请求，页面渲染配置请求包括至少一个样式渲染标识和渲染顺序，以便后续根据样式渲染标识生成原生底层接口或公共渲染代码，并根据渲染顺序对原生底层接口和公共渲染代码进行封装，以快速获取公共渲染接口，以便后续基于公共渲染接口进行页面渲染。基于样式渲染标识查询渲染函数库，判断渲染函数库中是否包含与样式渲染标识对应的渲染函数，渲染函数包括至少一个形式参数，以便后续查找到与样式渲染标识对应的渲染函数，将与样式渲染标识对应的实际参数赋值给渲染函数的形式参数，以实现渲染函数的赋值。若渲染函数库中包含与样式渲染标识对应的渲染函数，则获取与渲染函数中的形式参数相对应的实际参数，生成与样式渲染标识对应的原生底层接口，以便后续调用原生底层接口进行页面渲染，以提高渲染速度。若渲染函数库中不包含与样式渲染标识对应的渲染函数，则获取用户配置的与样式渲染标识对应的公共渲染代码，以便后续调用公共渲染代码进行页面渲染，以实现页面进行公共处理，减少渲染时间。根据渲染顺序，对原生底层接口和公共渲染代码进行封装，获取公共渲染接口，以便后续调用公共渲染接口，根据渲染顺序对页面进行渲染。

在一实施例中，若公共渲染代码封装有原生底层接口，通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面，具体包括如下步骤：若公共渲染代码封装有原生底层接口，则采用解析引擎对原生底层接口进行解析，获取与原生底层接口对应的渲染函数；根据渲染函数进行页面渲染，获取目标渲染页面。

其中，解析引擎可用于为 JavaScript 代码提供可扩展的运行环境，解析引擎具有扩展性，其可以提供各种灵活的扩展函数接口。具体而言，解析引擎可用于解析 JavaScript 代码中原生渲染接口，根据渲染函数库不同，解析引擎对应不同。在实际应用中，解析引擎具体可以包括：OPENGLES2.0、Nitro、V8 和 SpiderMonkey 等，本实施例中对具体的解析引擎不加以限制。

具体地，服务器获取到公共渲染代码，当公共渲染代码中封装有原生底层接口时，通过解析引擎解析原生底层接口，获取与原生底层接口对应的渲染函数，并根据渲染函数进行页面渲染，获取目标渲染页面。本实施例中，可实现对原生底层接口进行解析，获取渲染函数，以实现通过渲染函数进行页面渲染，避免不必要的运算，提高渲染速度。

应理解，上述实施例中各步骤的序号的大小并不意味着执行顺序的先后，各过程的执

行顺序应以其功能和内在逻辑确定，而不应对本申请实施例的实施过程构成任何限定。

在一实施例中，提供一种页面渲染装置，该页面渲染装置与上述实施例中页面渲染方法一一对应。如图 7 所示，该页面渲染装置包括请求获取模块 10、代码获取模块 20、底层接口渲染模块 30 和代码渲染模块 40。各功能模块详细说明如下：

请求获取模块 10，用于获取页面访问请求，页面访问请求包括 url 地址。

代码获取模块 20，用于根据 url 地址，获取与 url 地址对应的页面执行代码，页面执行代码包括至少一个公共渲染接口，每一公共渲染接口对应一公共渲染代码。

底层接口渲染模块 30，用于若公共渲染代码封装有原生底层接口，则通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面。

代码渲染模块 40，用于若公共渲染代码未封装有原生底层接口，则公共渲染接口调用公共渲染代码进行页面渲染，获取目标渲染页面。

在一实施例中，页面执行代码包括主页面执行代码和内容页面执行代码。代码获取模块 20 包括地址判断单元 21、第一代码获取单元 22 和第二代码获取单元 23。

地址判断单元 21，用于判断页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址。

第一代码获取单元 22，用于若对应同一 ip 地址，则根据 url 地址，获取与 url 地址对应的内容页面执行代码。

第二代码获取单元 23，用于若不对应同一 ip 地址，则根据 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执行代码。

在一实施例中，在第二代码获取单元 23 之后，页面渲染装置还包括主页面和内容页面渲染单元 2311 和第二页面渲染单元 2312。

主页面和内容页面渲染单元 2311，用于根据并行处理方式分别对主页面执行代码和内容页面执行代码进行页面渲染，分别获取主页面和内容页面。

第二页面渲染单元 2312，用于根据叠加方式对主页面和内容页面进行叠加，获取目标渲染页面。

在一实施例中，在第二代码获取单元 23 之后，页面渲染装置还包括公共渲染接口判断单元 2321、跳转单元 2322 和第三页面渲染单元 2323。

公共渲染接口判断单元 2321，用于判断 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口。

跳转单元 2322，用于若存在一致的公共渲染接口，则跳过公共渲染接口；

第三页面渲染单元 2323，用于若不存在一致的公共渲染接口，则基于公共渲染接口进行页面渲染，获取目标渲染页面。

在一实施例中，在底层接口渲染单元 21 之前，页面渲染装置还包括请求获取单元 201、渲染函数判断单元 202、原生底层接口生成单元 203、公共渲染代码获取单元 204 和公共渲染接口获取单元 205。

请求获取单元 201，用于获取页面渲染配置请求，页面渲染配置请求包括至少一个样式渲染标识和渲染顺序。

渲染函数判断单元 202，用于基于样式渲染标识查询渲染函数库，判断渲染函数库中

是否包含与样式渲染标识对应的渲染函数，渲染函数包括至少一个形式参数。

原生底层接口生成单元 203，用于若渲染函数库中包含与样式渲染标识对应的渲染函数，则获取与渲染函数中的形式参数相对应的实际参数，生成与样式渲染标识对应的原生底层接口。

公共渲染代码获取单元 204，用于若渲染函数库中不包含与样式渲染标识对应的渲染函数，则获取用户配置的与样式渲染标识对应的公共渲染代码。

公共渲染接口获取单元 205，用于根据渲染顺序，对原生底层接口和公共渲染代码进行封装，获取公共渲染接口。

在一实施例中，底层接口渲染单元 21 包括接口解析单元 211 和第一页面渲染单元 212。

接口解析单元 211，用于若公共渲染代码封装有原生底层接口，则采用解析引擎对原生底层接口进行解析，获取与原生底层接口对应的渲染函数。

第一页面渲染单元 212，用于根据渲染函数进行页面渲染，获取目标渲染页面。

关于页面渲染装置的具体限定可以参见上文中对于页面渲染方法的限定，在此不再赘述。上述页面渲染装置中的各个模块可全部或部分通过软件、硬件及其组合来实现。上述各模块可以硬件形式内嵌于或独立于计算机设备中的处理器中，也可以以软件形式存储于计算机设备中的存储器中，以便于处理器调用执行以上各个模块对应的操作。

在一个实施例中，提供了一种计算机设备，该计算机设备可以是服务器，其内部结构图可以如图 8 所示。该计算机设备包括通过系统总线连接的处理器、存储器、网络接口和数据库。其中，该计算机设备的处理器用于提供计算和控制能力。该计算机设备的存储器包括非易失性存储介质、内存储器。该非易失性存储介质存储有操作系统、计算机可读指令和数据库。该内存储器为易失性存储介质中的操作系统和计算机可读指令的运行提供环境。该计算机设备的数据库用于存储执行代码等。该计算机设备的网络接口用于与外部的终端通过网络连接通信。该计算机可读指令被处理器执行时以实现一种页面渲染方法。

在一个实施例中，提供了一种计算机设备，包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机可读指令，处理器执行计算机可读指令时实现以下步骤：获取页面访问请求，页面访问请求包括 url 地址；根据 url 地址，获取与 url 地址对应的页面执行代码，页面执行代码包括至少一个公共渲染接口，每一公共渲染接口对应一公共渲染代码；若公共渲染代码封装有原生底层接口，则通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面；若公共渲染代码未封装有原生底层接口，则公共渲染接口调用公共渲染代码进行页面渲染，获取目标渲染页面。

在一实施例中，页面执行代码包括主页面执行代码和内容页面执行代码；根据 url 地址，获取与 url 地址对应的页面执行代码，包括：判断页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址；若对应同一 ip 地址，则根据 url 地址，获取与 url 地址对应的内容页面执行代码；若不对应同一 ip 地址，则根据 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执行代码。

在一实施例中，在根据 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后，处理器执行计算机可读指令时还实现以下步骤：根据并行处理方式分别对主页面执行代码和内容页面执行代码进行页面渲染，分别获取主页面和内容页面；根

据叠加方式对主页面和内容页面进行叠加，获取目标渲染页面。

在一实施例中，在根据 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后，处理器执行计算机可读指令时还实现以下步骤：判断 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口；若存在一致的公共渲染接口，则跳过公共渲染接口；若不存在一致的公共渲染接口，则基于公共渲染接口进行页面渲染，获取目标渲染页面。

在一实施例中，若公共渲染代码封装有原生底层接口，则通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面的步骤之前，处理器执行计算机可读指令时实现以下步骤：获取页面渲染配置请求，页面渲染配置请求包括至少一个样式渲染标识和渲染顺序；基于样式渲染标识查询渲染函数库，判断渲染函数库中是否包含与样式渲染标识对应的渲染函数，渲染函数包括至少一个形式参数；若渲染函数库中包含与样式渲染标识对应的渲染函数，则获取与渲染函数中的形式参数相对应的实际参数，生成与样式渲染标识对应的原生底层接口；若渲染函数库中不包含与样式渲染标识对应的渲染函数，则获取用户配置的与样式渲染标识对应的公共渲染代码；根据渲染顺序，对原生底层接口和公共渲染代码进行封装，获取公共渲染接口。

在一实施例中，若公共渲染代码封装有原生底层接口，通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面，包括：若公共渲染代码封装有原生底层接口，则采用解析引擎对原生底层接口进行解析，获取与原生底层接口对应的渲染函数；根据渲染函数进行页面渲染，获取目标渲染页面。

在一个实施例中，提供一个或多个存储有计算机可读指令的非易失性可读存储介质，计算机可读存储介质存储有计算机可读指令，计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器执行如下步骤：获取页面访问请求，页面访问请求包括 url 地址；根据 url 地址，获取与 url 地址对应的页面执行代码，页面执行代码包括至少一个公共渲染接口，每一公共渲染接口对应一公共渲染代码；若公共渲染代码封装有原生底层接口，则通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面；若公共渲染代码未封装有原生底层接口，则公共渲染接口调用公共渲染代码进行页面渲染，获取目标渲染页面。

在一实施例中，页面执行代码包括主页面执行代码和内容页面执行代码；根据 url 地址，获取与 url 地址对应的页面执行代码，包括：判断页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址；若对应同一 ip 地址，则根据 url 地址，获取与 url 地址对应的内容页面执行代码；若不对应同一 ip 地址，则根据 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执行代码。

在一实施例中，在根据 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后，计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器还执行如下步骤：根据并行处理方式分别对主页面执行代码和内容页面执行代码进行页面渲染，分别获取主页面和内容页面；根据叠加方式对主页面和内容页面进行叠加，获取目标渲染页面。

在一实施例中，在根据 url 地址，获取与 url 地址对应的主页面执行代码和内容页面执

行代码的步骤之后，计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器还执行如下步骤：判断 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口；若存在一致的公共渲染接口，则跳过公共渲染接口；若不存在一致的公共渲染接口，则基于公共渲染接口进行页面渲染，获取目标渲染页面。

在一实施例中，若公共渲染代码封装有原生底层接口，则通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面的步骤之前，计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器执行如下步骤：获取页面渲染配置请求，页面渲染配置请求包括至少一个样式渲染标识和渲染顺序；基于样式渲染标识查询渲染函数库，判断渲染函数库中是否包含与样式渲染标识对应的渲染函数，渲染函数包括至少一个形式参数；若渲染函数库中包含与样式渲染标识对应的渲染函数，则获取与渲染函数中的形式参数相对应的实际参数，生成与样式渲染标识对应的原生底层接口；若渲染函数库中不包含与样式渲染标识对应的渲染函数，则获取用户配置的与样式渲染标识对应的公共渲染代码；根据渲染顺序，对原生底层接口和公共渲染代码进行封装，获取公共渲染接口。

在一实施例中，若公共渲染代码封装有原生底层接口，通过公共渲染接口调用原生底层接口进行页面渲染，获取目标渲染页面，包括：若公共渲染代码封装有原生底层接口，则采用解析引擎对原生底层接口进行解析，获取与原生底层接口对应的渲染函数；根据渲染函数进行页面渲染，获取目标渲染页面。

本领域普通技术人员可以理解实现上述实施例方法中的全部或部分流程，是可以通过计算机可读指令来指令相关的硬件来完成，计算机可读指令可存储于一非易失性计算机可读存储介质中，该计算机可读指令在执行时，可包括如上述各方法的实施例的流程。其中，本实施例中所提供的各实施例中所使用的对存储器、存储、数据库或其它介质的任何引用，均可包括非易失性和/或易失性存储器。非易失性存储器可包括只读存储器（ROM）、可编程 ROM（PROM）、电可编程 ROM（EPROM）、电可擦除可编程 ROM（EEPROM）或闪存。易失性存储器可包括随机存取存储器（RAM）或者外部高速缓冲存储器。作为说明而非局限，RAM 以多种形式可得，诸如静态 RAM（SRAM）、动态 RAM（DRAM）、同步 DRAM（SDRAM）、双数据率 SDRAM（DDRSDRAM）、增强型 SDRAM（ESDRAM）、同步链路（Synchlink）DRAM（SLDRAM）、存储器总线（Rambus）直接 RAM（RDRAM）、直接存储器总线动态 RAM（DRDRAM）、以及存储器总线动态 RAM（RDRAM）等。

所属领域的技术人员可以清楚地了解到，为了描述的方便和简洁，仅以上述各功能单元、模块的划分进行举例说明，实际应用中，可以根据需要而将上述功能分配由不同的功能单元、模块完成，即将装置的内部结构划分成不同的功能单元或模块，以完成以上描述的全部或者部分功能。

以上实施例仅用以说明本申请的技术方案，而非对其限制；尽管参照前述实施例对本申请进行了详细的说明，本领域的普通技术人员应当理解：其依然可以对前述各实施例所记载的技术方案进行修改，或者对其中部分技术特征进行等同替换；而这些修改或者替换，并不使相应技术方案的本质脱离本申请各实施例技术方案的精神和范围，均应包含在本申请的保护范围之内。

权 利 要 求 书

1. 一种页面渲染方法，其特征在于，包括：

获取页面访问请求，所述页面访问请求包括 url 地址；

根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，所述页面执行代码包括至少一个公共渲染接口，每一所述公共渲染接口对应一公共渲染代码；

若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面；

若所述公共渲染代码未封装有原生底层接口，则所述公共渲染接口调用所述公共渲染代码进行页面渲染，获取目标渲染页面。

2. 如权利要求 1 所述的页面渲染方法，其特征在于，所述页面执行代码包括主页面执行代码和内容页面执行代码；

所述根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，包括：

判断所述页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址；

若对应同一 ip 地址，则根据所述 url 地址，获取与所述 url 地址对应的内容页面执行代码；

若不对应同一 ip 地址，则根据所述 url 地址，获取与所述 url 地址对应的主页面执行代码和内容页面执行代码。

3. 如权利要求 2 所述的页面渲染方法，其特征在于，在所述根据所述 url 地址，获取与所述 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后，所述页面渲染方法还包括：

根据并行处理方式分别对所述主页面执行代码和所述内容页面执行代码进行页面渲染，分别获取主页面和内容页面；

根据叠加方式对所述主页面和所述内容页面进行叠加，获取目标渲染页面。

4. 如权利要求 2 所述的页面渲染方法，其特征在于，在所述根据所述 url 地址，获取与所述 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后，所述页面渲染方法还包括：

判断所述 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口；

若存在一致的公共渲染接口，则跳过所述公共渲染接口；

若不存在一致的公共渲染接口，则基于所述公共渲染接口进行页面渲染，获取目标渲染页面。

5. 如权利要求 1 所述的页面渲染方法，其特征在于，在所述若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面的步骤之前，所述页面渲染方法还包括：

获取页面渲染配置请求，所述页面渲染配置请求包括至少一个样式渲染标识和渲染顺序；

基于所述样式渲染标识查询渲染函数库，判断所述渲染函数库中是否包含与所述样式

渲染标识对应的渲染函数，所述渲染函数包括至少一个形式参数；

若所述渲染函数库中包含与所述样式渲染标识对应的渲染函数，则获取与渲染函数中的形式参数相对应的实际参数，生成与所述样式渲染标识对应的原生底层接口；

若所述渲染函数库中不包含与所述样式渲染标识对应的渲染函数，则获取用户配置的与所述样式渲染标识对应的公共渲染代码；

根据所述渲染顺序，对所述原生底层接口和所述公共渲染代码进行封装，获取公共渲染接口。

6. 如权利要求 1 所述的页面渲染方法，其特征在于，所述若所述公共渲染代码封装有原生底层接口，通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面，包括：

若所述公共渲染代码封装有原生底层接口，则采用解析引擎对所述原生底层接口进行解析，获取与所述原生底层接口对应的渲染函数；

根据所述渲染函数进行页面渲染，获取目标渲染页面。

7. 一种页面渲染装置，其特征在于，包括：

请求获取模块，用于获取页面访问请求，所述页面访问请求包括 url 地址；

代码获取模块，用于根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，所述页面执行代码包括至少一个公共渲染接口，每一所述公共渲染接口对应一公共渲染代码；

底层接口渲染模块，用于若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面；

代码渲染模块，用于若所述公共渲染代码未封装有原生底层接口，则所述公共渲染接口调用所述公共渲染代码进行页面渲染，获取目标渲染页面。

8. 如权利要求 7 所述的页面渲染装置，其特征在于，所述页面执行代码包括主页面执行代码和内容页面执行代码；

所述代码获取模块，包括：

地址判断单元，用于判断所述页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址；

第一代码获取单元，用于若对应同一 ip 地址，则根据所述 url 地址，获取与所述 url 地址对应的内容页面执行代码；

第二代码获取单元，用于若不对应同一 ip 地址，则根据所述 url 地址，获取与所述 url 地址对应的主页面执行代码和内容页面执行代码。

9. 一种计算机设备，包括存储器、处理器以及存储在所述存储器中并可在所述处理器上运行的计算机可读指令，其特征在于，所述处理器执行所述计算机可读指令时实现如下步骤：

获取页面访问请求，所述页面访问请求包括 url 地址；

根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，所述页面执行代码包括至少一个公共渲染接口，每一所述公共渲染接口对应一公共渲染代码；

若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底

层接口进行页面渲染，获取目标渲染页面；

若所述公共渲染代码未封装有原生底层接口，则所述公共渲染接口调用所述公共渲染代码进行页面渲染，获取目标渲染页面。

10. 如权利要求 9 所述的计算机设备，其特征在于，所述页面执行代码包括主页面执行代码和内容页面执行代码；

所述根据所述 url 地址，获取与所述 url 地址对应的页面执行代码，包括：

判断所述页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址；

若对应同一 ip 地址，则根据所述 url 地址，获取与所述 url 地址对应的内容页面执行代码；

若不对应同一 ip 地址，则根据所述 url 地址，获取与所述 url 地址对应的主页面执行代码和内容页面执行代码。

11. 如权利要求 10 所述的计算机设备，其特征在于，在所述根据所述 url 地址，获取与所述 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后，所述处理器执行所述计算机可读指令时还实现如下步骤：

根据并行处理方式分别对所述主页面执行代码和所述内容页面执行代码进行页面渲染，分别获取主页面和内容页面；

根据叠加方式对所述主页面和所述内容页面进行叠加，获取目标渲染页面。

12. 如权利要求 10 所述的计算机设备，其特征在于，在所述根据所述 url 地址，获取与所述 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后，所述处理器执行所述计算机可读指令时实现如下步骤：

判断所述 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口；

若存在一致的公共渲染接口，则跳过所述公共渲染接口；

若不存在一致的公共渲染接口，则基于所述公共渲染接口进行页面渲染，获取目标渲染页面。

13. 如权利要求 9 所述的计算机设备，其特征在于，在所述若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面的步骤之前，所述处理器执行所述计算机可读指令时实现如下步骤：

获取页面渲染配置请求，所述页面渲染配置请求包括至少一个样式渲染标识和渲染顺序；

基于所述样式渲染标识查询渲染函数库，判断所述渲染函数库中是否包含与所述样式渲染标识对应的渲染函数，所述渲染函数包括至少一个形式参数；

若所述渲染函数库中包含与所述样式渲染标识对应的渲染函数，则获取与渲染函数中的形式参数相对应的实际参数，生成与所述样式渲染标识对应的原生底层接口；

若所述渲染函数库中不包含与所述样式渲染标识对应的渲染函数，则获取用户配置的与所述样式渲染标识对应的公共渲染代码；

根据所述渲染顺序，对所述原生底层接口和所述公共渲染代码进行封装，获取公共渲染接口。

14. 如权利要求 9 所述的计算机设备, 其特征在于, 所述若所述公共渲染代码封装有原生底层接口, 通过所述公共渲染接口调用所述原生底层接口进行页面渲染, 获取目标渲染页面, 包括:

若所述公共渲染代码封装有原生底层接口, 则采用解析引擎对所述原生底层接口进行解析, 获取与所述原生底层接口对应的渲染函数;

根据所述渲染函数进行页面渲染, 获取目标渲染页面。

15. 一个或多个存储有计算机可读指令的非易失性可读存储介质, 所述计算机可读存储介质存储有计算机可读指令, 其特征在于, 所述计算机可读指令被一个或多个处理器执行时, 使得所述一个或多个处理器执行如下步骤:

获取页面访问请求, 所述页面访问请求包括 url 地址;

根据所述 url 地址, 获取与所述 url 地址对应的页面执行代码, 所述页面执行代码包括至少一个公共渲染接口, 每一所述公共渲染接口对应一公共渲染代码;

若所述公共渲染代码封装有原生底层接口, 则通过所述公共渲染接口调用所述原生底层接口进行页面渲染, 获取目标渲染页面;

若所述公共渲染代码未封装有原生底层接口, 则所述公共渲染接口调用所述公共渲染代码进行页面渲染, 获取目标渲染页面。

16. 如权利要求 15 所述的非易失性可读存储介质, 其特征在于, 所述页面执行代码包括主页面执行代码和内容页面执行代码;

所述根据所述 url 地址, 获取与所述 url 地址对应的页面执行代码, 包括:

判断所述页面访问请求的 url 地址与当前页面的 url 地址是否对应同一 ip 地址;

若对应同一 ip 地址, 则根据所述 url 地址, 获取与所述 url 地址对应的内容页面执行代码;

若不对应同一 ip 地址, 则根据所述 url 地址, 获取与所述 url 地址对应的主页面执行代码和内容页面执行代码。

17. 如权利要求 16 所述的非易失性可读存储介质, 其特征在于, 在所述根据所述 url 地址, 获取与所述 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后, 所述计算机可读指令被一个或多个处理器执行时, 使得所述一个或多个处理器还执行如下步骤:

根据并行处理方式分别对所述主页面执行代码和所述内容页面执行代码进行页面渲染, 分别获取主页面和内容页面;

根据叠加方式对所述主页面和所述内容页面进行叠加, 获取目标渲染页面。

18. 如权利要求 16 所述的非易失性可读存储介质, 其特征在于, 在所述根据所述 url 地址, 获取与所述 url 地址对应的主页面执行代码和内容页面执行代码的步骤之后, 所述计算机可读指令被一个或多个处理器执行时, 使得所述一个或多个处理器还执行如下步骤:

判断所述 url 地址对应的主页面执行代码中的公共渲染接口是否在当前页面的页面执行代码中存在一致的公共渲染接口;

若存在一致的公共渲染接口, 则跳过所述公共渲染接口;

若不存在一致的公共渲染接口，则基于所述公共渲染接口进行页面渲染，获取目标渲染页面。

19. 如权利要求 15 所述的非易失性可读存储介质，其特征在于，在所述若所述公共渲染代码封装有原生底层接口，则通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面的步骤之前，所述计算机可读指令被一个或多个处理器执行时，使得所述一个或多个处理器还执行如下步骤：

获取页面渲染配置请求，所述页面渲染配置请求包括至少一个样式渲染标识和渲染顺序；

基于所述样式渲染标识查询渲染函数库，判断所述渲染函数库中是否包含与所述样式渲染标识对应的渲染函数，所述渲染函数包括至少一个形式参数；

若所述渲染函数库中包含与所述样式渲染标识对应的渲染函数，则获取与渲染函数中的形式参数相对应的实际参数，生成与所述样式渲染标识对应的原生底层接口；

若所述渲染函数库中不包含与所述样式渲染标识对应的渲染函数，则获取用户配置的与所述样式渲染标识对应的公共渲染代码；

根据所述渲染顺序，对所述原生底层接口和所述公共渲染代码进行封装，获取公共渲染接口。

20. 如权利要求 15 所述的非易失性可读存储介质，其特征在于，所述若所述公共渲染代码封装有原生底层接口，通过所述公共渲染接口调用所述原生底层接口进行页面渲染，获取目标渲染页面，包括：

若所述公共渲染代码封装有原生底层接口，则采用解析引擎对所述原生底层接口进行解析，获取与所述原生底层接口对应的渲染函数；

根据所述渲染函数进行页面渲染，获取目标渲染页面。

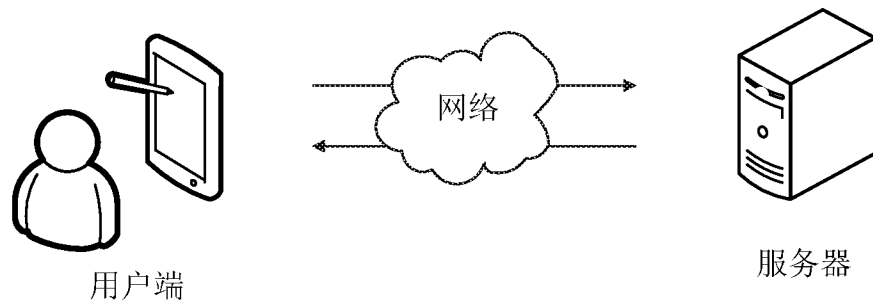


图 1

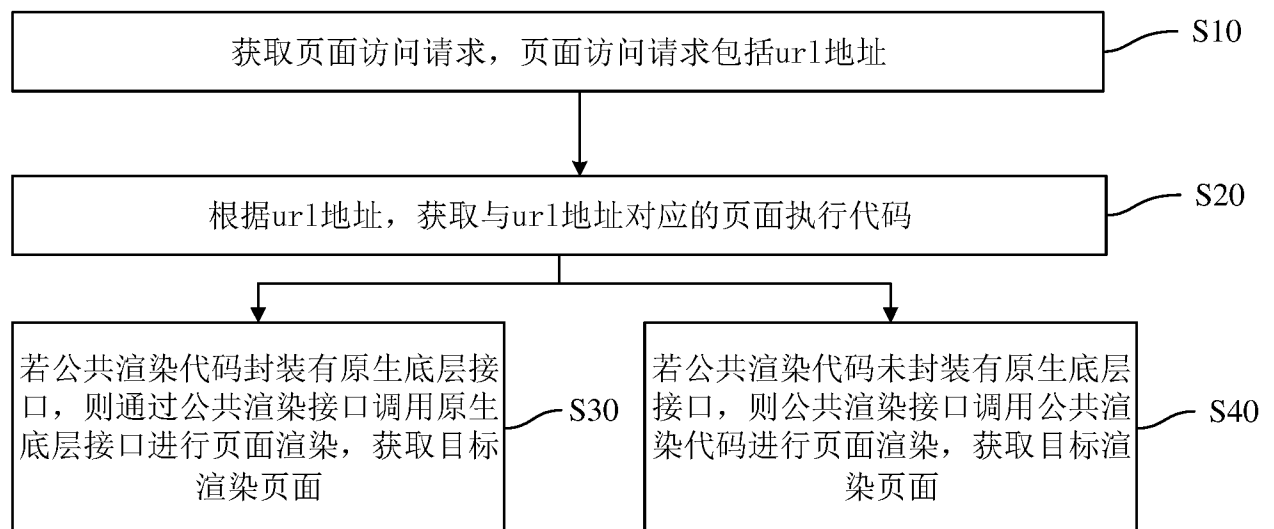


图 2

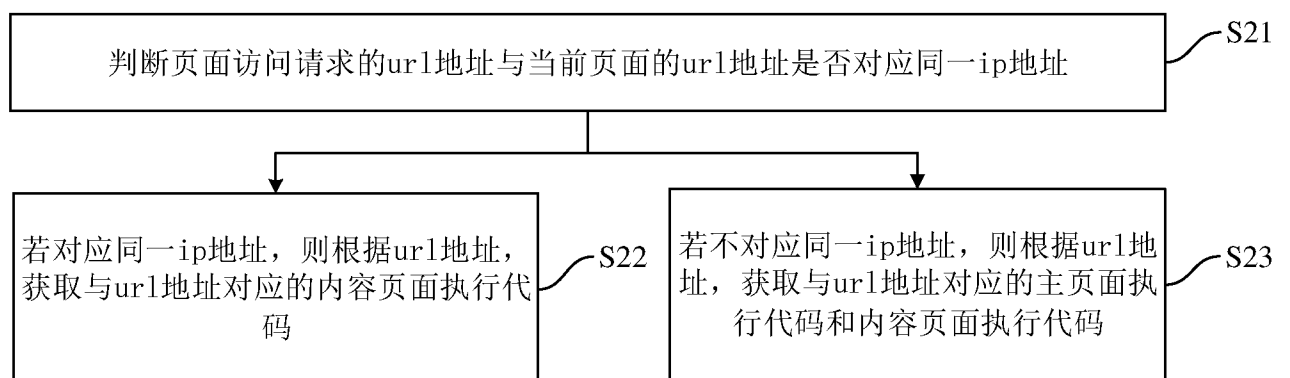
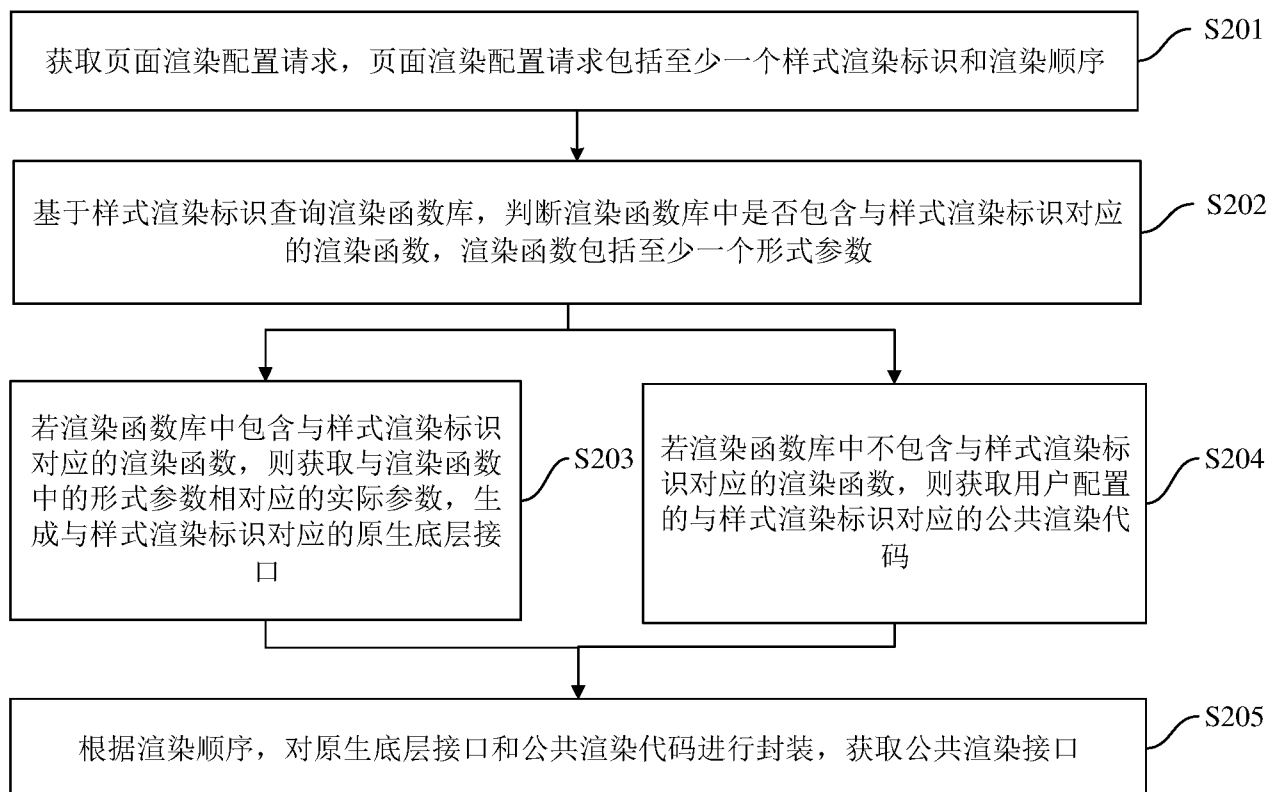
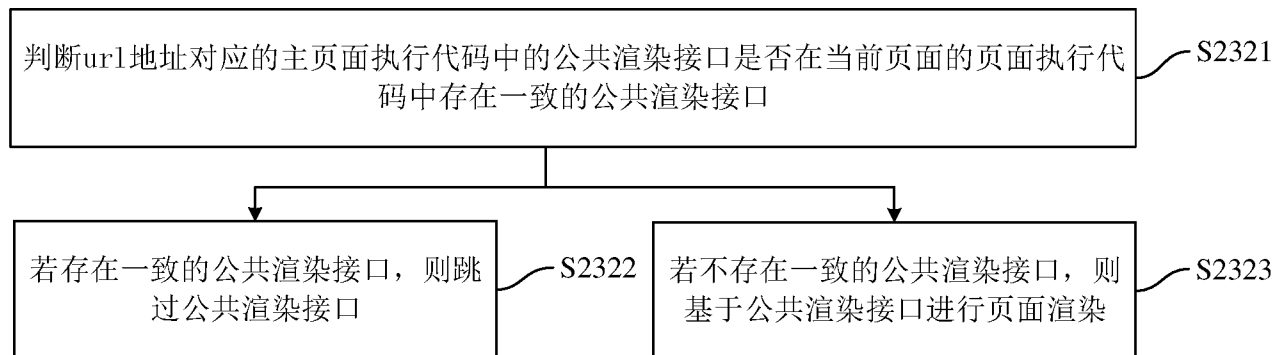
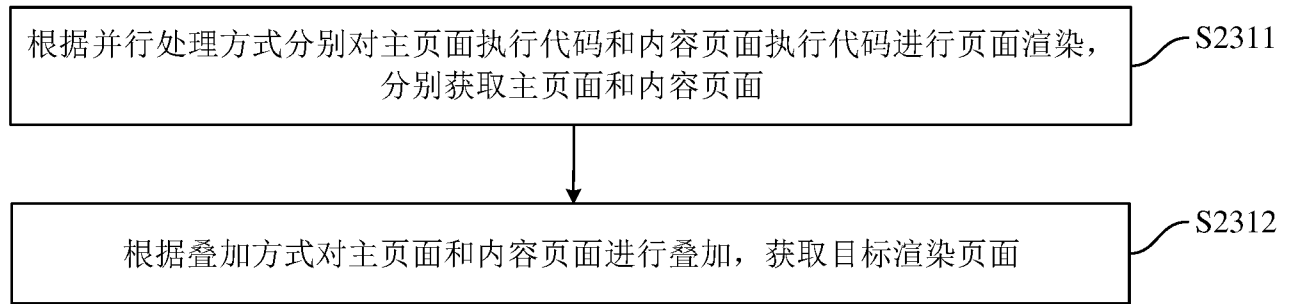


图 3



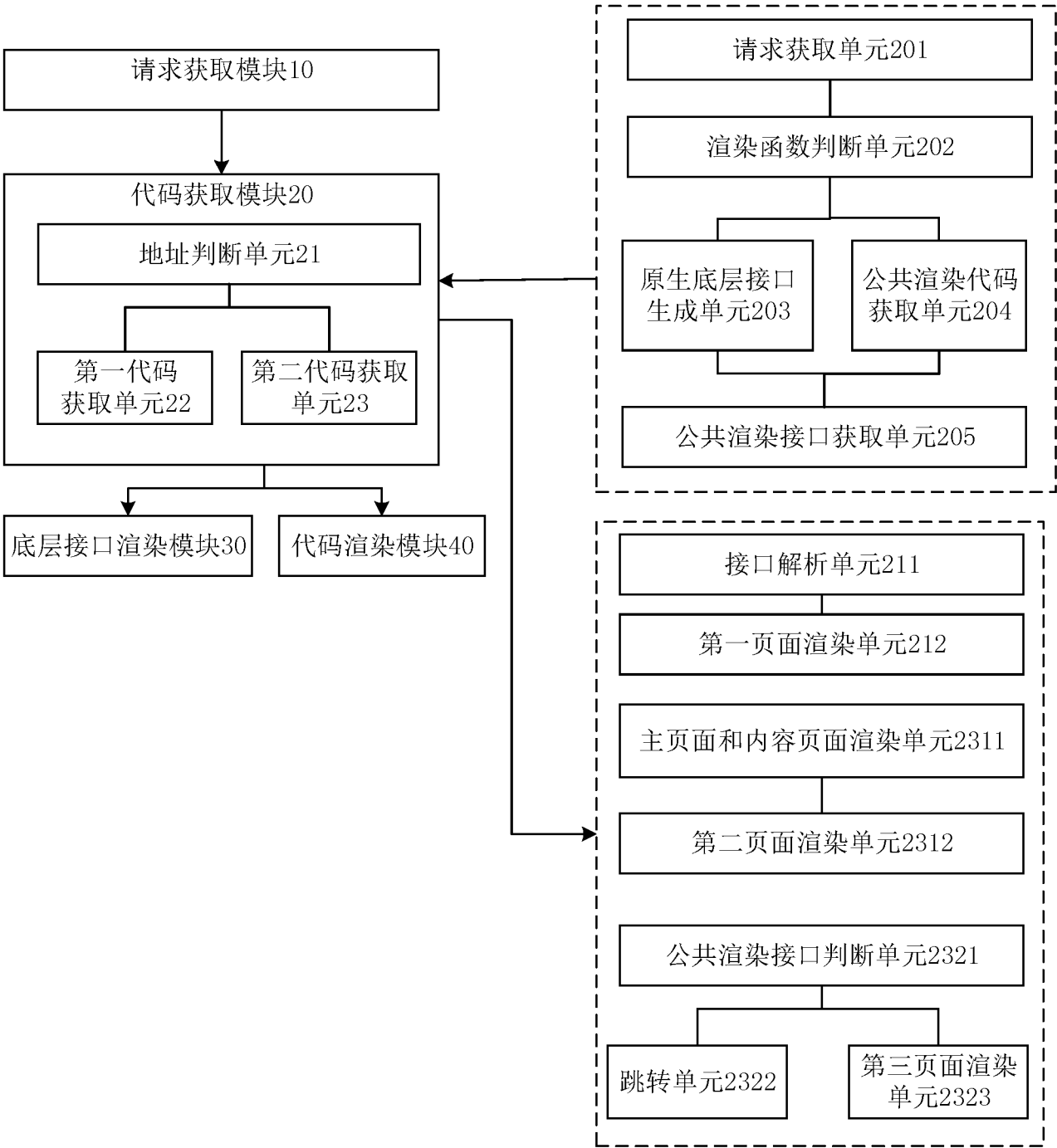


图 7

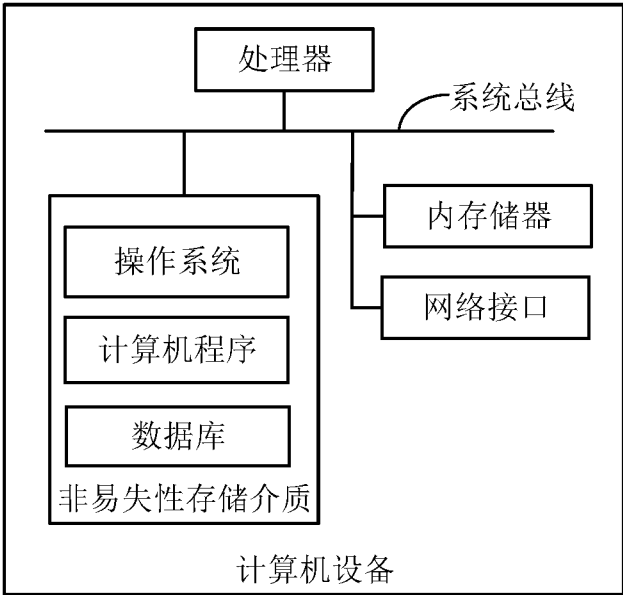


图 8

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2018/106434

A. CLASSIFICATION OF SUBJECT MATTER

G06F 16/957(2019.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT, CNKI, WPI, EPODOC: 页面, 网页, 渲染, 加载, 访问, 请求, url, 地址, 接口, 函数, 代码, 公共, 底层, 原生, 调用, webpage, render, load, query, address, interface, function, code, common, bottom, invoke

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 104020993 A (BEIJING KINGSOFT NETWORK TECHNOLOGY CO., LTD.) 03 September 2014 (2014-09-03) description, paragraphs [0003]-[0022]	1-20
X	CN 103955548 A (BEIJING KINGSOFT NETWORK TECHNOLOGY CO., LTD.) 30 July 2014 (2014-07-30) description, paragraphs [0003]-[0024]	1-20
A	CN 103544209 A (SHENZHEN TEMOBI SCIENCE & TECH DEVELOPMENT CO., LTD.) 29 January 2014 (2014-01-29) entire document	1-20
A	CN 102779164 A (SHENZHEN VISPRACTICE TECHNOLOGY CORPORATION) 14 November 2012 (2012-11-14) entire document	1-20
A	CN 103440151 A (BEIJING JINGDONG SHANGKE INFORMATION TECHNOLOGY CO., LTD.) 11 December 2013 (2013-12-11) entire document	1-20



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

“A” document defining the general state of the art which is not considered to be of particular relevance

“E” earlier application or patent but published on or after the international filing date

“L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

“O” document referring to an oral disclosure, use, exhibition or other means

“P” document published prior to the international filing date but later than the priority date claimed

“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

“X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

“Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

“&” document member of the same patent family

Date of the actual completion of the international search

19 April 2019

Date of mailing of the international search report

30 April 2019

Name and mailing address of the ISA/CN

State Intellectual Property Office of the P. R. China
No. 6, Xitucheng Road, Jimenqiao Haidian District, Beijing
100088
China

Facsimile No. (86-10)62019451

Authorized officer

Telephone No.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2018/106434

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN	104020993	A	03 September 2014	None	
CN	103955548	A	30 July 2014	None	
CN	103544209	A	29 January 2014	None	
CN	102779164	A	14 November 2012	None	
CN	103440151	A	11 December 2013	None	

国际检索报告

国际申请号

PCT/CN2018/106434

A. 主题的分类 G06F 16/957(2019.01) i 按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类																				
B. 检索领域 检索的最低限度文献(标明分类系统和分类号) G06F 包含在检索领域中的除最低限度文献以外的检索文献 在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用)) CNPAT、CNKI、WPI、EP0D0C: 页面、网页、渲染、加载、访问、请求、url、地址、接口、函数、代码、公共、底层、原生、调用、webpage、render、load、query、address、interface、function、code、common、bottom、invoke																				
C. 相关文件 <table border="1"> <thead> <tr> <th>类 型*</th> <th>引用文件, 必要时, 指明相关段落</th> <th>相关的权利要求</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>CN 104020993 A (北京金山网络科技有限公司) 2014年 9月 3日 (2014 - 09 - 03) 说明书第[0003]-[0022]段</td> <td>1-20</td> </tr> <tr> <td>X</td> <td>CN 103955548 A (北京金山网络科技有限公司) 2014年 7月 30日 (2014 - 07 - 30) 说明书第[0003]-[0024]段</td> <td>1-20</td> </tr> <tr> <td>A</td> <td>CN 103544209 A (深圳市融创天下科技股份有限公司) 2014年 1月 29日 (2014 - 01 - 29) 全文</td> <td>1-20</td> </tr> <tr> <td>A</td> <td>CN 102779164 A (深圳市远行科技有限公司) 2012年 11月 14日 (2012 - 11 - 14) 全文</td> <td>1-20</td> </tr> <tr> <td>A</td> <td>CN 103440151 A (北京京东尚科信息技术有限公司) 2013年 12月 11日 (2013 - 12 - 11) 全文</td> <td>1-20</td> </tr> </tbody> </table>			类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求	X	CN 104020993 A (北京金山网络科技有限公司) 2014年 9月 3日 (2014 - 09 - 03) 说明书第[0003]-[0022]段	1-20	X	CN 103955548 A (北京金山网络科技有限公司) 2014年 7月 30日 (2014 - 07 - 30) 说明书第[0003]-[0024]段	1-20	A	CN 103544209 A (深圳市融创天下科技股份有限公司) 2014年 1月 29日 (2014 - 01 - 29) 全文	1-20	A	CN 102779164 A (深圳市远行科技有限公司) 2012年 11月 14日 (2012 - 11 - 14) 全文	1-20	A	CN 103440151 A (北京京东尚科信息技术有限公司) 2013年 12月 11日 (2013 - 12 - 11) 全文	1-20
类 型*	引用文件, 必要时, 指明相关段落	相关的权利要求																		
X	CN 104020993 A (北京金山网络科技有限公司) 2014年 9月 3日 (2014 - 09 - 03) 说明书第[0003]-[0022]段	1-20																		
X	CN 103955548 A (北京金山网络科技有限公司) 2014年 7月 30日 (2014 - 07 - 30) 说明书第[0003]-[0024]段	1-20																		
A	CN 103544209 A (深圳市融创天下科技股份有限公司) 2014年 1月 29日 (2014 - 01 - 29) 全文	1-20																		
A	CN 102779164 A (深圳市远行科技有限公司) 2012年 11月 14日 (2012 - 11 - 14) 全文	1-20																		
A	CN 103440151 A (北京京东尚科信息技术有限公司) 2013年 12月 11日 (2013 - 12 - 11) 全文	1-20																		
☐ 其余文件在C栏的续页中列出。 ☒ 见同族专利附件。																				
* 引用文件的具体类型: “A” 认为不特别相关的表示了现有技术一般状态的文件 “E” 在国际申请日的当天或之后公布的在先申请或专利 “L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件(如具体说明的) “O” 涉及口头公开、使用、展览或其他方式公开的文件 “P” 公布日先于国际申请日但迟于所要求的优先权日的文件 “T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了理解发明之理论或原理的在后文件 “X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性 “Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性 “&” 同族专利的文件																				
国际检索实际完成的日期 2019年 4月 19日		国际检索报告邮寄日期 2019年 4月 30日																		
ISA/CN的名称和邮寄地址 中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088 传真号 (86-10)62019451		授权官员 赵晓敏 电话号码 86-(10)-53961332																		

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2018/106434

检索报告引用的专利文件			公布日 (年/月/日)	同族专利	公布日 (年/月/日)
CN	104020993	A	2014年 9月 3日	无	
CN	103955548	A	2014年 7月 30日	无	
CN	103544209	A	2014年 1月 29日	无	
CN	102779164	A	2012年 11月 14日	无	
CN	103440151	A	2013年 12月 11日	无	