



(12)发明专利申请

(10)申请公布号 CN 111090578 A

(43)申请公布日 2020.05.01

(21)申请号 201910969973.9

(22)申请日 2019.10.12

(71)申请人 中国平安人寿保险股份有限公司  
地址 518000 广东省深圳市福田区益田路  
5033号平安金融中心14、15、16、37、  
41、44、45、46层

(72)发明人 蔡堂健

(74)专利代理机构 深圳市精英专利事务所  
44242

代理人 李翔宇

(51)Int.Cl.  
G06F 11/36(2006.01)

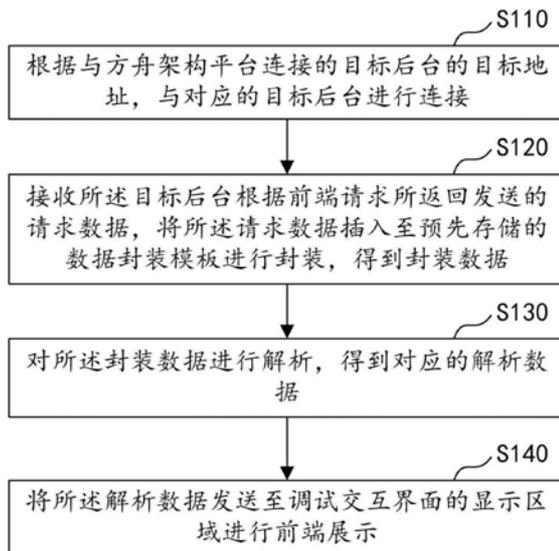
权利要求书2页 说明书9页 附图7页

(54)发明名称

工程开发联调方法、装置、计算机设备及存储介质

(57)摘要

本发明公开了工程开发联调方法、装置、计算机设备及存储介质。该方法包括根据与方舟架构平台连接的目标后台的目标地址，与对应的目标后台进行连接；其中，所述目标地址包括目标后台的IP地址及端口号；接收所述目标后台根据前端请求所返回发送的请求数据，将所述请求数据插入至预先存储的数据封装模板进行封装，得到封装数据；对所述封装数据进行解析，得到对应的解析数据；将所述解析数据发送至调试交互界面的显示区域进行前端展示。该方法实现了由目标后台的目标地址直接与目标后台进行连接进行直连，在指定了目标后台后即可准确的访问到目标后台从而进行联调，提高了联调的效率。



1. 一种工程开发联调方法,其特征在于,包括:

根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接;其中,所述目标地址包括目标后台的IP地址及端口号;

接收所述目标后台根据前端请求所返回发送的请求数据,将所述请求数据插入至预先存储的数据封装模板进行封装,得到封装数据;

对所述封装数据进行解析,得到对应的解析数据;以及

将所述解析数据发送至调试交互界面的显示区域进行前端展示。

2. 根据权利要求1所述的工程开发联调方法,其特征在于,所述根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接之前,还包括:

获取与方舟架构平台连接的初始配置文件;

接收目标后台的目标地址,将所述初始配置文件中的地址信息以所述目标地址进行替换,得到当前配置文件;

所述根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接,包括:

根据所述当前配置文件与对应的目标后台进行连接。

3. 根据权利要求1所述的工程开发联调方法,其特征在于,所述根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接之前,还包括:

获取本地的历史请求数据,定位所述历史请求数据中与预设关键词相同的第一数据结构体;

将所述第一数据结构体中的数据进行删除,得到数据封装模板。

4. 根据权利要求1所述的工程开发联调方法,其特征在于,所述根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接之前,还包括:

获取方舟架构平台中的交互数据,定位所述交互数据中与预设关键词相同的第二数据结构体;

将所述第二数据结构体中的数据进行删除,得到数据封装模板。

5. 根据权利要求1所述的工程开发联调方法,其特征在于,所述对所述封装数据进行解析,得到对应的解析数据,包括:

定位获取所述封装数据的数据结构体;

获取所述数据结构体中的数据,以得到解析数据。

6. 根据权利要求1所述的工程开发联调方法,其特征在于,所述将所述解析数据发送至调试交互界面的显示区域进行前端展示之后,还包括:

若检测到确认编辑指令,将对所述解析数据的编辑信息发送至目标后台。

7. 一种工程开发联调装置,其特征在于,包括:

后台直连单元,用于根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接;其中,所述目标地址包括目标后台的IP地址及端口号;

数据封装单元,用于接收所述目标后台根据前端请求所返回发送的请求数据,将所述请求数据插入至预先存储的数据封装模板进行封装,得到封装数据;

数据解析单元,用于对所述封装数据进行解析,得到对应的解析数据;以及

数据显示单元,用于将所述解析数据发送至调试交互界面的显示区域进行前端展示。

8. 根据权利要求7所述的工程开发联调装置,其特征在于,还包括:  
初始配置文件获取单元,用于获取与方舟架构平台连接的初始配置文件;  
地址替换单元,用于接收目标后台的目标地址,将所述初始配置文件中的地址信息以所述目标地址进行替换,得到当前配置文件;  
所述后台直连单元,还用于:  
根据所述当前配置文件与对应的目标后台进行连接。
9. 一种计算机设备,包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序,其特征在于,所述处理器执行所述计算机程序时实现如权利要求1至6中任一项所述的工程开发联调方法。
10. 一种计算机可读存储介质,其特征在于,所述计算机可读存储介质存储有计算机程序,所述计算机程序当被处理器执行时使所述处理器执行如权利要求1至6任一项所述的工程开发联调方法。

## 工程开发联调方法、装置、计算机设备及存储介质

### 技术领域

[0001] 本发明涉及功能测试技术领域,尤其涉及一种工程开发联调方法、装置、计算机设备及存储介质。

### 背景技术

[0002] 目前使用的ARK方舟架构中,在开发阶段进行前后台工程联调(联调是指前端页面和后台服务之间的接口调试)时,后台工程启动时会将服务部署到方舟架构平台中。前端页面通过方舟架构平台访问后台服务会存在随机访问的问题,即前端页面多次访问同一接口而得到不同的后台服务反馈结果,而不是指定了接口之后就一定访问到对应的后台服务,降低了联调的效率。具体的,即一般来说一个后台工程会有多个前端工程师参与开发不同模块,当多个后台同时启动进行调试的时候,前台访问服务时将会随机访问某个后台服务,而可能无法访问到想要访问的服务,给联调带来很大的困难。

### 发明内容

[0003] 本发明实施例提供了一种工程开发联调方法、装置、计算机设备及存储介质,旨在解决现有技术中在ARK方舟架构中前端页面通过方舟架构平台访问后台服务会存在随机访问的问题,即前端页面多次访问同一接口而得到不同的后台服务反馈结果,降低了联调的效率的问题。

[0004] 第一方面,本发明实施例提供了一种工程开发联调方法,其包括:

[0005] 根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接;其中,所述目标地址包括目标后台的IP地址及端口号;

[0006] 接收所述目标后台根据前端请求所返回发送的请求数据,将所述请求数据插入至预先存储的数据封装模板进行封装,得到封装数据;

[0007] 对所述封装数据进行解析,得到对应的解析数据;以及

[0008] 将所述解析数据发送至调试交互界面的显示区域进行前端展示。

[0009] 第二方面,本发明实施例提供了一种工程开发联调装置,其包括:

[0010] 后台直连单元,用于根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接;其中,所述目标地址包括目标后台的IP地址及端口号;

[0011] 数据封装单元,用于接收所述目标后台根据前端请求所返回发送的请求数据,将所述请求数据插入至预先存储的数据封装模板进行封装,得到封装数据;

[0012] 数据解析单元,用于对所述封装数据进行解析,得到对应的解析数据;以及

[0013] 数据显示单元,用于将所述解析数据发送至调试交互界面的显示区域进行前端展示。

[0014] 第三方面,本发明实施例又提供了一种计算机设备,其包括存储器、处理器及存储在所述存储器上并可在所述处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现上述第一方面所述的工程开发联调方法。

[0015] 第四方面,本发明实施例还提供了一种计算机可读存储介质,其中所述计算机可读存储介质存储有计算机程序,所述计算机程序当被处理器执行时使所述处理器执行上述第一方面所述的工程开发联调方法。

[0016] 本发明实施例提供了一种工程开发联调方法、装置、计算机设备及存储介质,实现了由目标后台的目标地址直接与目标后台进行连接进行直连,在指定了目标后台后即可准确的访问到目标后台从而进行联调,提高了联调的效率。

## 附图说明

[0017] 为了更清楚地说明本发明实施例技术方案,下面将对实施例描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0018] 图1为本发明实施例提供的工程开发联调方法的应用场景示意图;

[0019] 图2为本发明实施例提供的工程开发联调方法的流程示意图;

[0020] 图3为本发明实施例提供的工程开发联调方法的另一流程示意图;

[0021] 图4为本发明实施例提供的工程开发联调方法的另一流程示意图;

[0022] 图5为本发明实施例提供的工程开发联调方法的另一流程示意图;

[0023] 图6为本发明实施例提供的工程开发联调方法的子流程示意图;

[0024] 图7为本发明实施例提供的工程开发联调装置的示意性框图;

[0025] 图8为本发明实施例提供的工程开发联调装置的另一示意性框图;

[0026] 图9为本发明实施例提供的工程开发联调装置的另一示意性框图;

[0027] 图10为本发明实施例提供的工程开发联调装置的另一示意性框图;

[0028] 图11为本发明实施例提供的工程开发联调装置的子单元示意性框图;

[0029] 图12为本发明实施例提供的计算机设备的示意性框图。

## 具体实施方式

[0030] 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。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。

[0031] 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。

[0032] 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。

[0033] 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。

[0034] 请参阅图1和图2,图1为本发明实施例提供的工程开发联调方法的应用场景示意图;图2为本发明实施例提供的工程开发联调方法的流程示意图,该工程开发联调方法应用

于服务器中,该方法通过安装于服务器中的应用软件进行执行。

[0035] 如图2所示,该方法包括步骤S110~S140。

[0036] S110、根据与方舟架构平台连接的目标后台的目标地址,与对应的目标后台进行连接;其中,所述目标地址包括目标后台的IP地址及端口号。

[0037] 在本实施例中,为了更清楚的理解本申请的技术方案,下面对所涉及的终端进行介绍。本申请是在前端的角度描述技术方案。

[0038] 一是ARK方舟构架平台,简记为方舟架构平台,用于前后端分离框架的项目,也可以说方舟架构就是一套微服务架构,类似于springcloud。

[0039] 二是ARK方舟构架平台的前端,是项目的前端开发人员使用而进行项目开发。

[0040] 三是ARK方舟架构平台的后台,是项目的后台开发人员使用而进行项目开发。

[0041] 一般ARK方舟构架平台中一个后台工程会有多个工程师参与开发不同模块,每一个模块可能还有不同时期的版本(例如V1.0,V2.0版本),若同一模块的不同版本分布在不同的后台上时,当这多个后台同时启动进行调试的时候,前台访问该模块的服务时将会随机访问该模块对应的某个后台服务,而可能无法访问到想要访问的服务,给联调带来很大的困难。例如,前端输入的访问模块A,则有可能接收到的反馈是来自后台A1、后台A2、后台A3其中一个,这就导致联调效率低下。此时为了实现前端与指定后台的直连,需要获知目标后台的地址信息,将所获知的目标后台的地址信息进行解析后,实现前端与对应的目标后台的直连。

[0042] 在一实施例中,如图3所示,步骤S110之前还包括:

[0043] S101、获取与方舟架构平台连接的初始配置文件;

[0044] S102、接收目标后台的目标地址,将所述初始配置文件中的地址信息以所述目标地址进行替换,得到当前配置文件。

[0045] 在一实施例中,步骤S110包括:

[0046] 根据所述当前配置文件与对应的目标后台进行连接。

[0047] 在本实施例中,例如,前端与方舟架构平台连接的初始配置文件为:

[0048] proxyTable: {

[0049] ' /admin' : {

[0050] target: ' http://aaaa.bbbb.com.cn'

[0051] pathRewrite: {

[0052] ' ^/admin' : '/'

[0053] },

[0054] },

[0055] }

[0056] 其中,target后的地址表示前端连接aaaa.bbbb.com.cn地址对应的ARK方舟构架平台后,由ARK方舟构架平台随机分配一个后台。

[0057] 当从目标后台获取了其目标地址信息后,可以将初始配置文件中target后的地址用目标地址信息进行替换,例如目标地址信息为10.110.110.11:6000,则将aaaa.bbbb.com.cn替换为10.110.110.11:6000即可得到当前配置文件,其中目标地址信息中10.110.110.11表示目标后台的IP地址,6000表示目标后台的端口号。当初始配置文件修

改为当前配置文件后,前端就可以指定与目标后台进行直连。

[0058] 但是由于之前后台与前端之间通信是基于ARK方舟构架平台的数据格式,若前端直接向与之直连的后台发送数据请求,后台所反馈回来的数据在前端是无法解析的,此时需进行数据后续的封装处理。

[0059] S120、接收所述目标后台根据前端请求所返回发送的请求数据,将所述请求数据插入至预先存储的数据封装模板进行封装,得到封装数据。

[0060] 在本实施例中,前端一般会预先存储有用于对目标后台根据前端请求所返回发送的请求数据进行格式转化的数据封装模板,以便于前端识别对应的请求数据。前端中所存储的数据封装模板,有至少两种获取方式,一种是从前端的本地的历史请求数据中提取,另一种是通过方舟架构平台中的交互数据中获取。

[0061] 在一实施例中,如图4所示,作为获取数据封装模板的第一实施例,步骤S110之前还包括:

[0062] S103、获取本地的历史请求数据,定位所述历史请求数据中与预设关键词相同的第一数据结构体;

[0063] S104、将所述第一数据结构体中的数据进行删除,得到数据封装模板。

[0064] 在本实施例中,历史请求数据即是前端通过ARK方舟构架平台与后端交互时在前端存储的历史数据,例如历史数据中有respond.data这一关键词,获取respond.data之后数据结构体内的数据内容,数据结构体的具体数据一般在{}内,例如respond.data {}即为所获得的数据封装模板。

[0065] 在一实施例中,如图6所示,作为获取数据封装模板的第二实施例,步骤S110之前还包括:

[0066] S105、获取方舟架构平台中的交互数据,定位所述交互数据中与预设关键词相同的第二数据结构体;

[0067] S106、将所述第二数据结构体中的数据进行删除,得到数据封装模板。

[0068] 在本实施例中,前端经过方舟架构平台与后台进行交互时,在方舟架构平台中缓存或永久存储有交互数据,从方舟架构平台中存储的多条交互数据中获取其中一条作为分析样本。例如交互数据中有respond.data这一关键词,获取respond.data之后数据结构体内的数据内容,数据结构体的具体数据一般在{}内,例如respond.data {}即为所获得的数据封装模板。

[0069] 通过上述两个获取数据封装模板的实施例,能有效提取数据封装模板以对请求数据进行封装,使得前端能有效识别请求数据。

[0070] S130、对所述封装数据进行解析,得到对应的解析数据。

[0071] 在本实施例中,对所述封装数据进行解析,即解析封装数据的数据结构体内的数据内容,从而得到后台反馈的数据。之前之所以将请求数据通过封装模板进行封装,是因为不经过这一处理,前端识别不出请求数据(因为前端原先只能识别经ARK方舟构架平台反馈的数据,请求数据由后台反馈至前端之前,是经过了ARK方舟构架平台的封装处理,此时将这一封装过程前置至前端的本地),将请求数据在前端的本地一个指定区域先通过封装模板封装得到封装数据后,才能被前端正常识别读取并解析。

[0072] 在一实施例中,如图6所示,步骤S130包括:

[0073] S131、定位获取所述封装数据的数据结构体；

[0074] S132、获取所述数据结构体中的数据，以得到解析数据。

[0075] 在本实施例中，对封装数据进行解析，即是封装过程的逆向过程。例如封装数据中有respond.data这一关键词，获取respond.data之后数据结构体内的数据内容，数据结构体的具体数据一般在{}内，例如respond.data {}中大括号之内的数据即为所获得的解析数据。通过对封装数据进行解析，即可获取与前端请求相对应的解析数据，以供在前端进行联调。

[0076] S140、将所述解析数据发送至调试交互界面的显示区域进行前端展示。

[0077] 在本实施例中，当前端接收了所述解析数据后（如模块A的V2.0版本的代码数据），此时为了便于前端的操作用户进行调试，需将所述解析数据发送至调试交互界面的显示区域进行前端展示。在前端是搭载了与后端中类似的调试平台，在调试平台的调试交互界面的显示区域对所述解析数据进行展示，即可便于前端的开发人员对解析数据进行查看或是编辑。

[0078] 在一实施例中，步骤S140之后还包括：

[0079] 若检测到确认编辑指令，将对所述解析数据的编辑信息发送至目标后台。

[0080] 在本实施例中，当前端的操作用户在查看所述解析数据后，可所述解析数据进行修改和编辑，在完成编辑并点击保存后即出发编辑确认指令，此时一旦前端检测到确认编辑指令，由前端将所述解析数据的编辑信息（即更新后的解析数据）发送至目标后台，以实现前端和后台的联调。

[0081] 该方法实现了由目标后台的目标地址直接与目标后台进行连接进行直连，在指定了目标后台后即可准确的访问到目标后台从而进行联调，提高了联调的效率。

[0082] 本发明实施例还提供一种工程开发联调装置，该工程开发联调装置用于执行前述工程开发联调方法的任一实施例。具体地，请参阅图7，图7是本发明实施例提供的工程开发联调装置的示意性框图。该工程开发联调装置100可以配置于服务器中。

[0083] 如图7所示，工程开发联调装置100包括后台直连单元110、数据封装单元120、数据解析单元130、数据显示单元140。

[0084] 后台直连单元110，用于根据与方舟架构平台连接的目标后台的目标地址，与对应的目标后台进行连接；其中，所述目标地址包括目标后台的IP地址及端口号。

[0085] 在本实施例中，一般ARK方舟构架平台中一个后台工程会有多个工程师参与开发不同模块，每一个模块可能还有不同时期的版本（例如V1.0，V2.0版本），若同一模块的不同版本分布在不同的后台上时，当这多个后台同时启动进行调试的时候，前台访问该模块的服务时将会随机访问该模块对应的某个后台服务，而可能无法访问到想要访问的服务，给联调带来很大的困难。例如，前端输入的访问模块A，则有可能接收到的反馈是来自后台A1、后台A2、后台A3其中一个，这就导致联调效率低下。此时为了实现前端与指定后台的直连，需要获知目标后台的地址信息，将所获知的目标后台的地址信息进行解析后，实现前端与对应的目标后台的直连。

[0086] 在一实施例中，如图8所示，工程开发联调装置100还包括：

[0087] 初始配置文件获取单元101，用于获取与方舟架构平台连接的初始配置文件；

[0088] 地址替换单元102，用于接收目标后台的目标地址，将所述初始配置文件中的地址

信息以所述目标地址进行替换,得到当前配置文件。

[0089] 在一实施例中,后台直连单元S110还用于:

[0090] 根据所述当前配置文件与对应的目标后台进行连接。

[0091] 在本实施例中,例如,前端与方舟架构平台连接的初始配置文件为:

```
[0092] proxyTable:{
[0093]   '/admin' :{
[0094]     target:' http://aaaa.bbbb.com.cn'
[0095]     pathRewrite:{
[0096]       '^/admin' : '/'
[0097]     },
[0098]   },
[0099] }
```

[0100] 其中,target后的地址表示前端连接aaaa.bbbb.com.cn地址对应的ARK方舟构架平台后,由ARK方舟构架平台随机分配一个后台。

[0101] 当从目标后台获取了其目标地址信息后,可以将初始配置文件中target后的地址用目标地址信息进行替换,例如目标地址信息为10.110.110.11:6000,则将aaaa.bbbb.com.cn替换为10.110.110.11:6000即可得到当前配置文件,其中目标地址信息中10.110.110.11表示目标后台的IP地址,6000表示目标后台的端口号。当初始配置文件修改为当前配置文件后,前端就可以指定与目标后台进行直连。

[0102] 但是由于之前后台与前端之间通信是基于ARK方舟构架平台的数据格式,若前端直接向与之直连的后台发送数据请求,后台所反馈回来的数据在前端是无法解析的,此时需进行数据后续的封装处理。

[0103] 数据封装单元120,用于接收所述目标后台根据前端请求所返回发送的请求数据,将所述请求数据插入至预先存储的数据封装模板进行封装,得到封装数据。

[0104] 在本实施例中,前端一般会预先存储有用于对目标后台根据前端请求所返回发送的请求数据进行格式转化的数据封装模板,以便于前端识别对应的请求数据。前端中所存储的数据封装模板,有至少两种获取方式,一种是从前端的本地的历史请求数据中提取,另一种是通过方舟架构平台中的交互数据中获取。

[0105] 在一实施例中,如图9所示,作为获取数据封装模板的第一实施例,工程开发联调装置100还包括:

[0106] 第一数据结构体获取单元103,用于获取本地的历史请求数据,定位所述历史请求数据中与预设关键词相同的第一数据结构体;

[0107] 第一数据删除单元104,用于将所述第一数据结构体中的数据进行删除,得到数据封装模板。

[0108] 在本实施例中,历史请求数据即是前端通过ARK方舟构架平台与后端交互时在前端存储的历史数据,例如历史数据中有respond.data这一关键词,获取respond.data之后数据结构体内的数据内容,数据结构体的具体数据一般在{}内,例如respond.data {}即为所获得的数据封装模板。

[0109] 在一实施例中,如图10所示,作为获取数据封装模板的第二实施例,工程开发联调

装置100还包括：

[0110] 第二数据结构体获取单元105,用于获取方舟架构平台中的交互数据,定位所述交互数据中与预设关键词相同的第二数据结构体；

[0111] 第二数据删除单元106,用于将所述第二数据结构体中的数据进行删除,得到数据封装模板。

[0112] 在本实施例中,前端经过方舟架构平台与后台进行交互时,在方舟架构平台中缓存或永久存储有交互数据,从方舟架构平台中存储的多条交互数据中获取其中一条作为分析样本。例如交互数据中有respond.data这一关键词,获取respond.data之后数据结构体内的数据内容,数据结构体的具体数据一般在{}内,例如respond.data{}即为所获得的数据封装模板。

[0113] 通过上述两个获取数据封装模板的实施例,能有效提取数据封装模板以对请求数据进行封装,使得前端能有效识别请求数据。

[0114] 数据解析单元130,用于对所述封装数据进行解析,得到对应的解析数据。

[0115] 在本实施例中,对所述封装数据进行解析,即解析封装数据的数据结构体内的数据内容,从而得到后台反馈的数据。之前之所以将请求数据通过封装模板进行封装,是因为不经过这一处理,前端识别不出请求数据(因为前端原先只能识别经ARK方舟构架平台反馈的数据,请求数据由后台反馈至前端之前,是经过了ARK方舟构架平台的封装处理,此时将这一封装过程前置至前端的本地),将请求数据在前端的本地一个指定区域先通过封装模板封装得到封装数据后,才能被前端正常识别读取并解析。

[0116] 在一实施例中,如图11所示,数据解析单元130包括：

[0117] 结构体定位单元131,用于定位获取所述封装数据的数据结构体；

[0118] 结构体数据提取单元132,用于获取所述数据结构体中的数据,以得到解析数据。

[0119] 在本实施例中,对封装数据进行解析,即是封装过程的逆向过程。例如封装数据中有respond.data这一关键词,获取respond.data之后数据结构体内的数据内容,数据结构体的具体数据一般在{}内,例如respond.data{}中大括号之内的数据即为所获得的解析数据。通过对封装数据进行解析,即可获取与前端请求相对应的解析数据,以供在前端进行联调。

[0120] 数据显示单元140,用于将所述解析数据发送至调试交互界面的显示区域进行前端展示。

[0121] 在本实施例中,当前端接收了所述解析数据后(如模块A的V2.0版本的代码数据),此时为了便于前端的操作用户进行调试,需将所述解析数据发送至调试交互界面的显示区域进行前端展示。在前端是搭载了与后端中类似的调试平台,在调试平台的调试交互界面的显示区域对所述解析数据进行展示,即可便于前端的开发人员对解析数据进行查看或是编辑。

[0122] 在一实施例中,工程开发联调装置100还包括：

[0123] 编辑数据发送单元,用于若检测到确认编辑指令,将对所述解析数据的编辑信息发送至目标后台。

[0124] 在本实施例中,当前端的操作用户在查看所述解析数据后,可所述解析数据进行修改和编辑,在完成编辑并点击保存后即出发编辑确认指令,此时一旦前端检测到确认编

辑指令,由前端将所述解析数据的编辑信息(即更新后的解析数据)发送至目标后台,以实现前端和后台的联调。

[0125] 该装置实现了由目标后台的目标地址直接与目标后台进行连接进行直连,在指定了目标后台后即可准确的访问到目标后台从而进行联调,提高了联调的效率。

[0126] 上述工程开发联调装置可以实现为计算机程序的形式,该计算机程序可以在如图12所示的计算机设备上运行。

[0127] 请参阅图12,图12是本发明实施例提供的计算机设备的示意性框图。该计算机设备500是服务器,服务器可以是独立的服务器,也可以是多个服务器组成的服务器集群。

[0128] 参阅图12,该计算机设备500包括通过系统总线501连接的处理器502、存储器和网络接口505,其中,存储器可以包括非易失性存储介质503和内存储器504。

[0129] 该非易失性存储介质503可存储操作系统5031和计算机程序5032。该计算机程序5032被执行时,可使得处理器502执行工程开发联调方法。

[0130] 该处理器502用于提供计算和控制能力,支撑整个计算机设备500的运行。

[0131] 该内存储器504为非易失性存储介质503中的计算机程序5032的运行提供环境,该计算机程序5032被处理器502执行时,可使得处理器502执行工程开发联调方法。

[0132] 该网络接口505用于进行网络通信,如提供数据信息的传输等。本领域技术人员可以理解,图12中示出的结构,仅仅是与本发明方案相关的部分结构的框图,并不构成对本发明方案所应用于其上的计算机设备500的限定,具体的计算机设备500可以包括比图中所示更多或更少的部件,或者组合某些部件,或者具有不同的部件布置。

[0133] 其中,所述处理器502用于运行存储在存储器中的计算机程序5032,以实现本发明实施例公开的工程开发联调方法。

[0134] 本领域技术人员可以理解,图12中示出的计算机设备的实施例并不构成对计算机设备具体构成的限定,在其他实施例中,计算机设备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。例如,在一些实施例中,计算机设备可以仅包括存储器及处理器,在这样的实施例中,存储器及处理器的结构及功能与图12所示实施例一致,在此不再赘述。

[0135] 应当理解,在本发明实施例中,处理器502可以是中央处理单元(Central Processing Unit,CPU),该处理器502还可以是其他通用处理器、数字信号处理器(Digital Signal Processor,DSP)、专用集成电路(Application Specific Integrated Circuit,ASIC)、现成可编程门阵列(Field-Programmable Gate Array,FPGA)或者其他可编程逻辑器件、分立门或者晶体管逻辑器件、分立硬件组件等。其中,通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。

[0136] 在本发明的另一实施例中提供计算机可读存储介质。该计算机可读存储介质可以为非易失性的计算机可读存储介质。该计算机可读存储介质存储有计算机程序,其中计算机程序被处理器执行时实现本发明实施例公开的工程开发联调方法。

[0137] 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,上述描述的设备、装置和单元的具体工作过程,可以参考前述方法实施例中的对应过程,在此不再赘述。本领域普通技术人员可以意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,能够以电子硬件、计算机软件或者二者的结合来实现,为了清楚地说明硬件和软件的

可互换性,在上述说明中已经按照功能一般性地描述了各示例的组成及步骤。这些功能究竟以硬件还是软件方式来执行取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本发明的范围。

[0138] 在本发明所提供的几个实施例中,应该理解到,所揭露的设备、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,所述单元的划分,仅仅为逻辑功能划分,实际实现时可以有另外的划分方式,也可以将具有相同功能的单元集成成一个单元,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。另外,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、装置或单元的间接耦合或通信连接,也可以是电的,机械的或其它的形式连接。

[0139] 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发明实施例方案的目的。

[0140] 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,也可以是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。

[0141] 所述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,可以存储在一个存储介质中。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,服务器,或者网络设备等)执行本发明各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(ROM,Read-Only Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0142] 以上所述,仅为本发明的具体实施方式,但本发明的保护范围并不局限于此,任何熟悉本技术领域的技术人员在本发明揭露的技术范围内,可轻易想到各种等效的修改或替换,这些修改或替换都应涵盖在本发明的保护范围之内。因此,本发明的保护范围应以权利要求的保护范围为准。

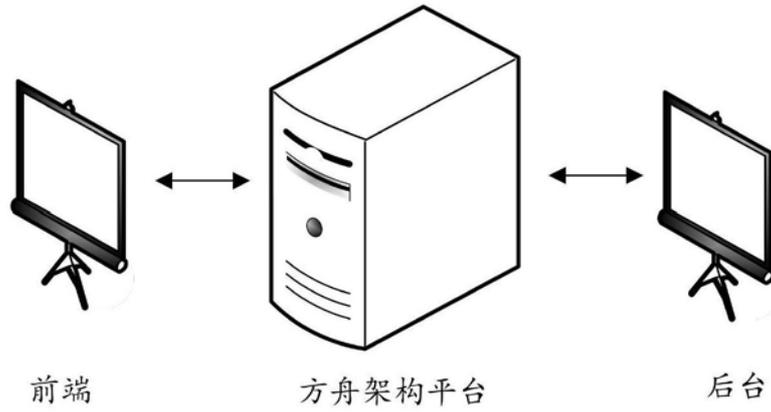


图1

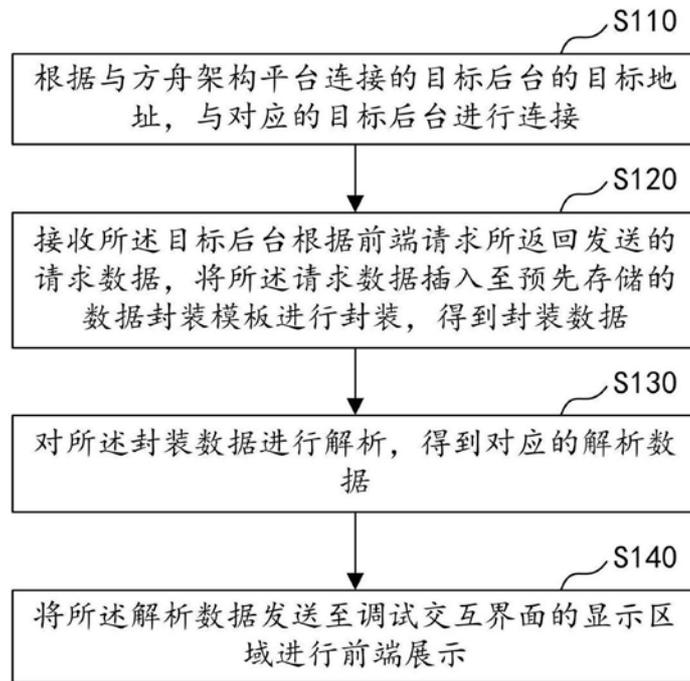


图2

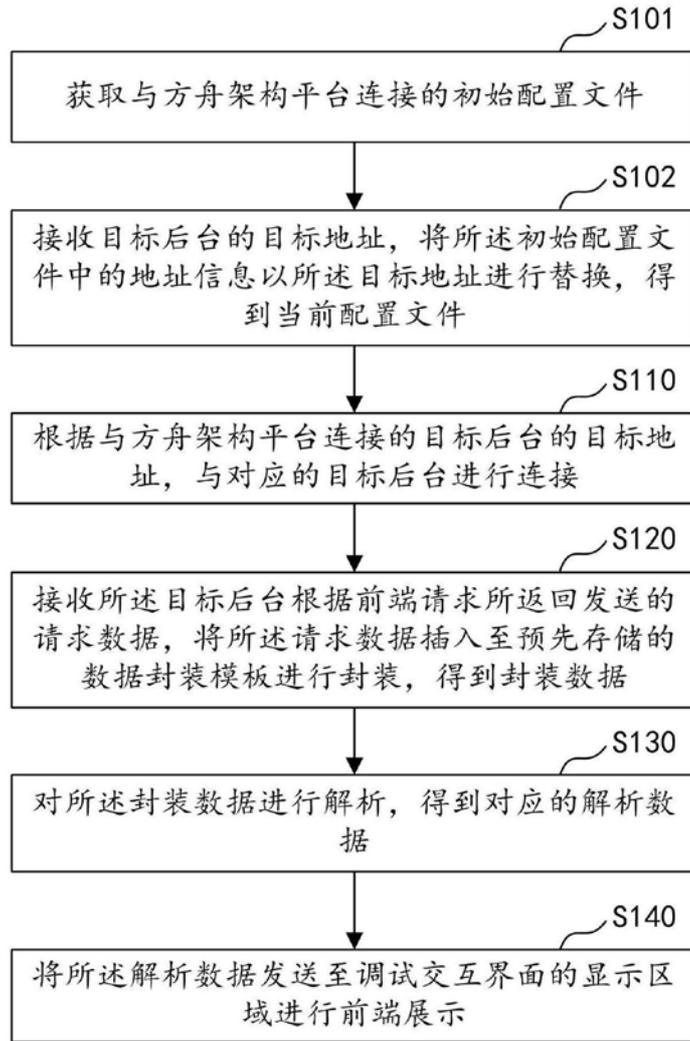


图3

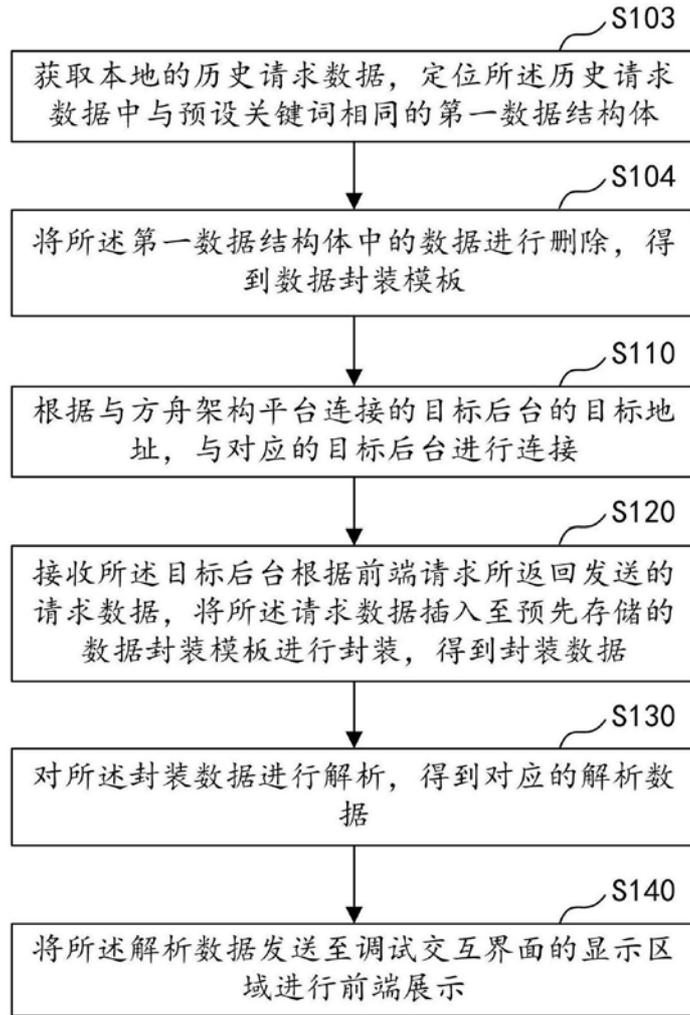


图4

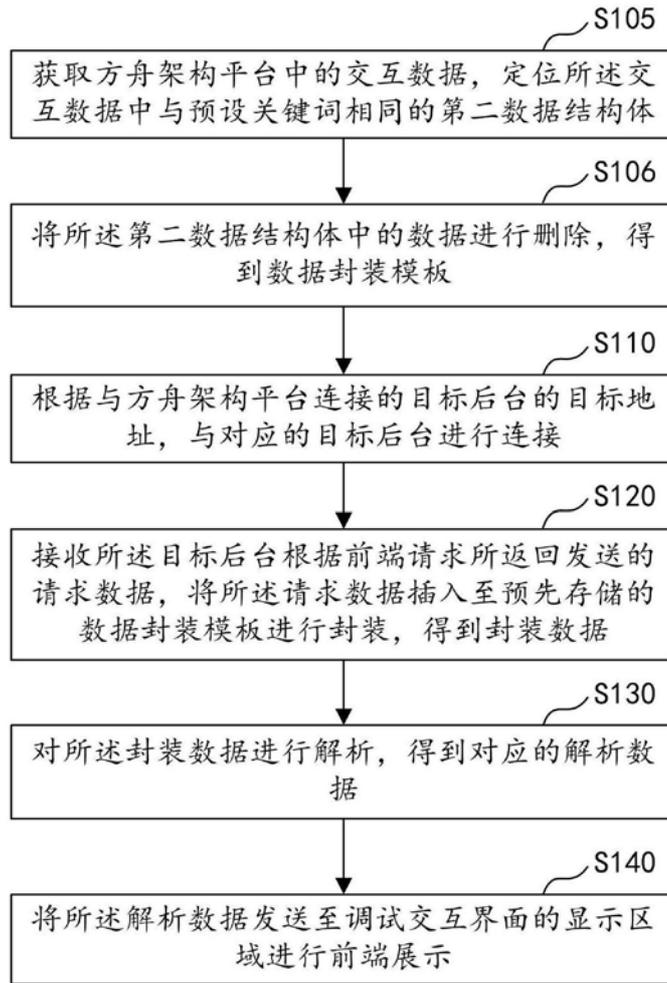


图5

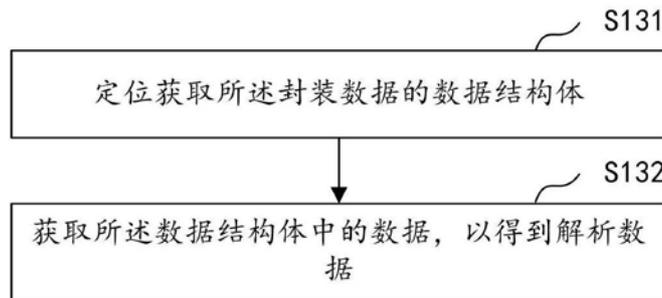


图6

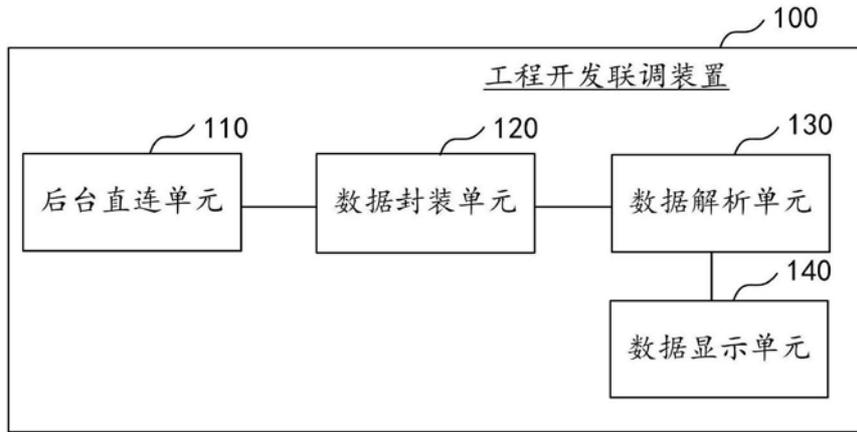


图7

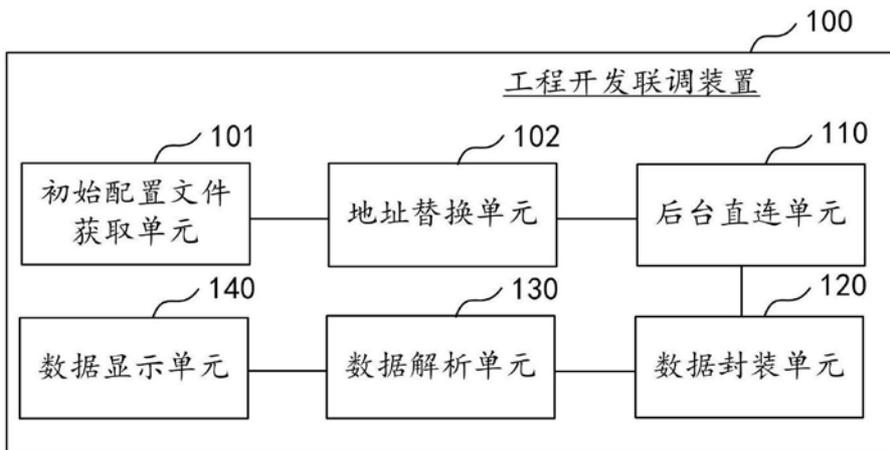


图8

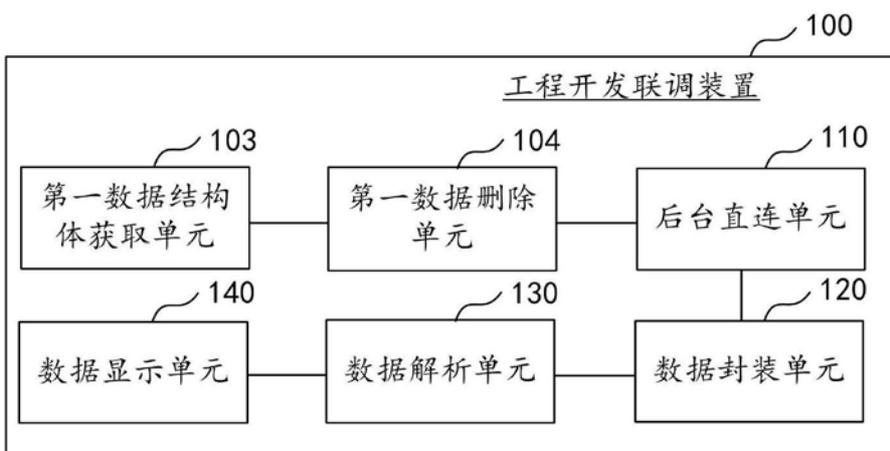


图9

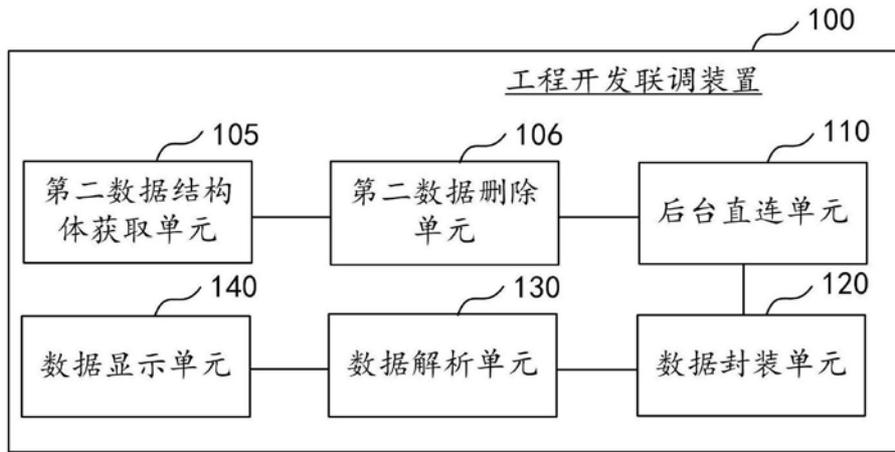


图10

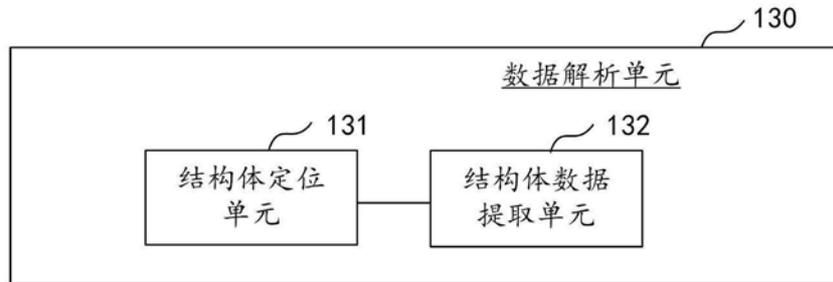


图11

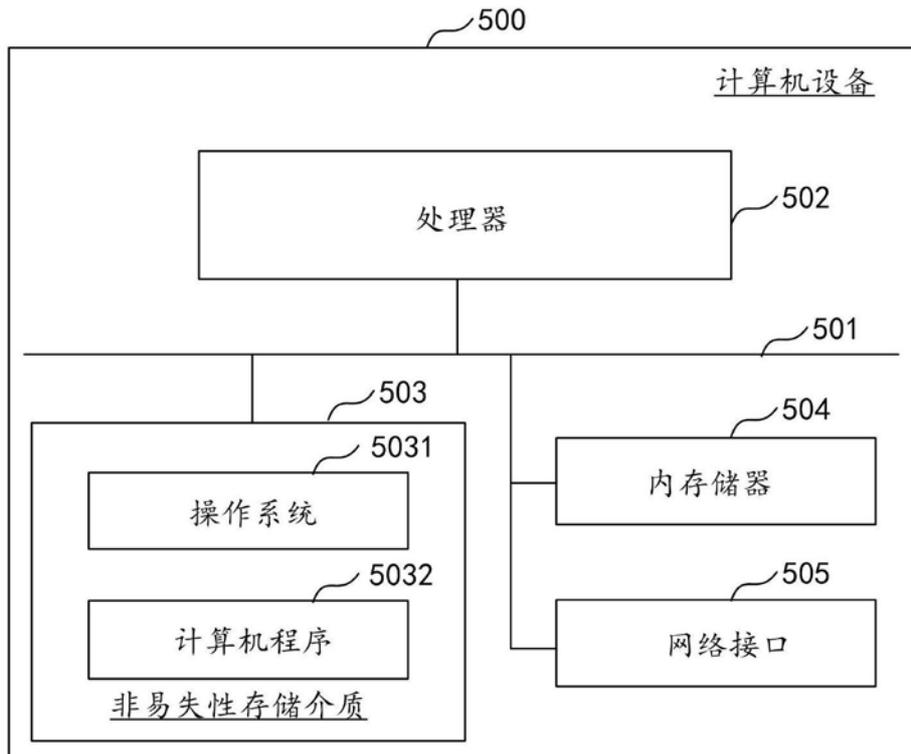


图12