



(12)发明专利申请

(10)申请公布号 CN 108153745 A

(43)申请公布日 2018.06.12

(21)申请号 201611096626.2

(22)申请日 2016.12.02

(71)申请人 腾讯科技(深圳)有限公司

地址 518057 广东省深圳市南山区高新区
科技中一路腾讯大厦35层

(72)发明人 梁宇轩

(74)专利代理机构 深圳市深佳知识产权代理事
务所(普通合伙) 44285

代理人 王仲凯

(51) Int. Cl.

G06F 17/30(2006.01)

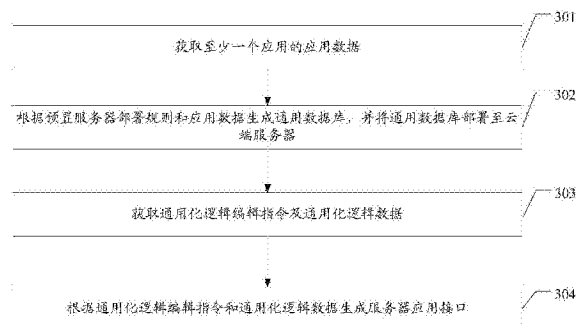
权利要求书2页 说明书9页 附图4页

(54)发明名称

一种应用数据调用方法及应用开发系统

(57)摘要

本发明公开了一种应用数据调用方法及应用开发系统,用于实现服务器部署及生成服务器应用接口,使得应用开发过程中,无需后端工程师处理通用化逻辑,从而节省了人力和时间的耗费,提高了开发效率。本发明实施方法包括:获取至少一个应用的应用数据;根据预置服务器部署规则和应用数据生成通用数据库,并将通用数据库部署至云端服务器;获取通用化逻辑编辑指令及通用化逻辑数据,通用化逻辑编辑指令用于编辑通用化逻辑数据,通用化逻辑数据用于表示处理应用数据的逻辑数据;根据通用化逻辑编辑指令和通用化逻辑生成服务器应用接口,服务器应用接口用于调用云端服务器中的应用数据。



1. 一种应用数据调用方法,其特征在于,包括:
 - 获取至少一个应用的应用数据;
 - 根据预置服务器部署规则和所述应用数据生成通用数据库,并将所述通用数据库部署至云端服务器;
 - 获取通用化逻辑编辑指令及通用化逻辑数据,所述通用化逻辑编辑指令用于编辑所述通用化逻辑数据,所述通用化逻辑数据用于表示处理所述应用数据的逻辑数据;
 - 根据所述通用化逻辑编辑指令和所述通用化逻辑生成服务器应用接口,所述服务器应用接口用于调用所述云端服务器中的所述应用数据。
2. 根据权利要求1所述的应用数据调用方法,其特征在于,所述根据所述通用化逻辑编辑指令和所述通用化逻辑数据生成服务器应用接口,包括:
 - 根据所述通用化逻辑编辑指令编辑所述通用化逻辑数据,得到逻辑字段,所述逻辑字段的字段类型为字符、文件或图片;
 - 对所述逻辑字段进行封装处理,生成服务器应用接口。
3. 根据权利要求2所述的应用数据调用方法,其特征在于,所述根据所述通用化逻辑编辑指令编辑所述通用化逻辑数据,得到逻辑字段之后,还包括:
 - 根据预置加密规则对所述逻辑字段进行加密,得到密钥信息;
 - 根据预置配置规则设置所述逻辑字段的访问权限信息;
 - 根据所述逻辑字段、所述密钥信息及所述访问权限信息生成逻辑数据表。
4. 根据权利要求1所述的应用数据调用方法,其特征在于,所述获取通用化逻辑编辑指令及通用化逻辑数据,包括:
 - 接收用户根据通用化逻辑输入的通用化逻辑编辑指令;
 - 根据所述通用化逻辑编辑指令确定对应的通用化逻辑;
 - 获取所述通用化逻辑的通用化逻辑数据。
5. 根据权利要求1至4中任一项所述的应用数据调用方法,其特征在于,所述根据预置服务器部署规则和所述应用数据生成通用数据库,并将所述通用数据库部署至云端服务器,包括:
 - 根据所述预置服务器部署规则从所述应用中确定通用化应用;
 - 从所述应用数据中选择出所述通用化应用的应用数据,并生成通用数据库,所述通用数据库包括用户权限表、可查询日志文件及连接密钥;
 - 根据所述预置服务器部署规则将所述通用数据库部署至云端服务器。
6. 一种应用开发系统,其特征在于,包括:
 - 获取模块,用于获取至少一个应用的应用数据;
 - 部署模块,用于根据预置服务器部署规则和所述应用数据生成通用数据库,并将所述通用数据库部署至云端服务器;
 - 获取模块,还用于获取通用化逻辑编辑指令及通用化逻辑数据,所述通用化逻辑编辑指令用于编辑所述通用化逻辑数据,所述通用化逻辑数据用于表示处理所述应用数据的逻辑数据;
 - 接口生成模块,用于根据所述通用化逻辑编辑指令和所述通用化逻辑数据生成服务器应用接口,所述服务器应用接口用于调用所述云端服务器中的所述应用数据。

7. 根据权利要求6所述的应用开发系统,其特征在于,
所述接口生成模块,具体用于根据所述通用化逻辑编辑指令编辑所述通用化逻辑数据,得到逻辑字段,所述逻辑字段的字段类型为字符、文件或图片;
所述接口生成模块,还用于对所述逻辑字段进行封装处理,生成服务器应用接口。
8. 根据权利要求7所述的应用开发系统,其特征在于,所述应用开发系统还包括:
数据处理模块,具体用于根据预置加密规则对所述逻辑字段进行加密,得到密钥信息;
所述数据处理模块,还用于根据预置配置规则设置所述逻辑字段的访问权限信息;
所述数据处理模块,还用于根据所述逻辑字段、所述密钥信息及所述访问权限信息生成逻辑数据表。
9. 根据权利要求6所述的应用开发系统,其特征在于,
所述获取模块,还用于接收用户根据通用化逻辑输入的通用化逻辑编辑指令;
所述获取模块,还用于根据所述通用化逻辑编辑指令确定对应的通用化逻辑;
所述获取模块,还用于获取所述通用化逻辑的通用化逻辑数据。
10. 根据权利要求6至9中任一项所述的应用开发系统,其特征在于,
所述部署模块,具体用于根据所述预置服务器部署规则从所述应用中确定通用化应用;
所述部署模块,还用于从所述应用数据中选择出所述通用化应用的应用数据,并生成通用数据库,所述通用数据库包括用户权限表、可查询日志文件及连接密钥;
所述部署模块,还用于根据所述预置服务器部署规则将所述通用数据库部署至云端服务器。

一种应用数据调用方法及应用开发系统

技术领域

[0001] 本发明涉及互联网技术领域,尤其涉及一种应用数据调用方法及应用开发系统。

背景技术

[0002] 随着移动端的发展、企业内部办公自动化 (Office Automation,OA) 系统的增加及搜索引擎优化 (Search Engine Optimization,SEO) 的弱化,传统意义上的后端数据开发在新时代的利用占比有所降低,而以ajax为核心的JSON (JavaScript Object Notation) 数据传输模式在移动端开发和企业内部OA等的利用率越来越高。

[0003] 移动端应用开发的数据传输模式基本以ajax为核心的JSON数据传输,数据的类型与数据传输的方式有很大的通用性,而从网页开发与应用开发的数据模块来说,90%的数据模块都是类似的,数据传输无非增、删、改及查等等。在常规的开发方案中,每套程序都会配置后端工程师对服务器进行配置和对程序逻辑数据进行手动书写程序。

[0004] 但是,在开发网页和应用等移动端程序时,后端工程师需要对每一个程序的服务器数据调用进行配置以及对程序逻辑进行处理,使得开发过程中人力和时间的耗费较大,影响了开发效率。

发明内容

[0005] 本发明提供了一种应用数据调用方法及应用开发系统,用于实现服务器部署及生成服务器应用接口,使得应用开发过程中,无需后端工程师处理通用化逻辑,从而节省了人力和时间的耗费,提高了开发效率。

[0006] 本发明第一方面提供一种应用数据调用方法,包括:

[0007] 获取至少一个应用的应用数据;

[0008] 根据预置服务器部署规则和所述应用数据生成通用数据库,并将所述通用数据库部署至云端服务器;

[0009] 获取通用化逻辑编辑指令及通用化逻辑数据,所述通用化逻辑编辑指令用于编辑所述通用化逻辑数据,所述通用化逻辑数据用于表示处理所述应用数据的逻辑数据;

[0010] 根据所述通用化逻辑编辑指令和所述通用化逻辑数据生成服务器应用接口,所述服务器应用接口用于调用所述云端服务器中的所述应用数据。

[0011] 本发明第二方面提供一种应用开发系统,包括:

[0012] 获取模块,用于获取至少一个应用的应用数据;

[0013] 部署模块,用于根据预置服务器部署规则和所述应用数据生成通用数据库,并将所述通用数据库部署至云端服务器;

[0014] 获取模块,还用于获取通用化逻辑编辑指令及通用化逻辑数据,所述通用化逻辑编辑指令用于编辑所述通用化逻辑数据,所述通用化逻辑数据用于表示处理所述应用数据的逻辑数据;

[0015] 接口生成模块,用于根据所述通用化逻辑编辑指令和所述通用化逻辑数据生成服

务器应用接口,所述服务器应用接口用于调用所述云端服务器中的所述应用数据。

[0016] 从以上技术方案可以看出,本发明实施例的应用数据调用方法和应用开发系统具有以下优点:

[0017] 应用开发系统获取至少一个应用的应用数据;根据预置服务器部署规则和应用数据生成通用数据库,并将通用数据库部署至云端服务器;获取通用化逻辑编辑指令及通用化逻辑数据,通用化逻辑编辑指令用于编辑通用化逻辑数据,通用化逻辑数据用于表示处理应用数据的逻辑数据;根据通用化逻辑编辑指令和通用化逻辑数据生成服务器应用接口,服务器应用接口用于调用云端服务器中的应用数据。由于应用数据库是根据预置服务器部署规则部署到云端服务器的,并且通用化逻辑处理是通过通用化逻辑编辑指令实现,前端设备通过生成的服务器应用接口就能进行调用应用数据,与现有技术相比,通用化逻辑不需要后端工程师对每一个通用化功能的服务器数据调用进行配置以及对通用化逻辑进行处理,因此,节省了人力和时间的耗费,提高了开发效率。

附图说明

[0018] 为了更清楚地说明本发明实施例技术方案,下面将对实施例和现有技术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它的附图。

[0019] 图1为现有的移动端应用开发的系统框架图;

[0020] 图2为本发明提供的移动端应用开发的系统框架图;

[0021] 图3为本发明实施例中应用数据调用方法的一个实施例示意图;

[0022] 图4为本发明实施例中根据逻辑字段生成服务器应用接口的流程示意图;

[0023] 图5为本发明实施例中应用开发系统的一个结构示意图;

[0024] 图6为本发明实施例中应用开发系统的另一个结构示意图;

[0025] 图7为本发明实施例中应用开发系统的一个实体装置实结构示意图。

具体实施方式

[0026] 本发明提供了一种应用数据调用方法及应用开发系统,用于实现服务器部署及生成服务器应用接口,使得应用开发过程中,无需后端工程师处理通用化逻辑,从而节省了人力和时间的耗费,提高了开发效率。

[0027] 为了使本技术领域的人员更好地理解本发明方案,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,所描述的实施例仅仅是本发明一部分的实施例,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,都应当属于本发明保护的范畴。

[0028] 如图1所示,为现有的移动端应用开发的系统框架图,应用开发人员开发出应用时,将该应用的应用数据存储到云端服务器,前端设备(即移动端)可以使用该应用时需要访问云端服务器的应用数据,但是云端服务器的应用数据不是所有的前端设备都可以访问,那么就需要部署服务器的数据调用,而且前端设备与云端服务器进行数据传输都是具

有逻辑的,后端工程师需要手动书写代码进行服务器部署和逻辑处理,在图1中,后端工程师以后端设备进行代替,因为后端工程师所进行的工作是需要针对应用开发、服务器部署和前端设备与云端服务器数据传输的。

[0029] 本发明应用于移动端应用中,应用开发的系统框架图如图2所示,其中应用开发系统可以是实体的设备(例如个人计算机、服务器或局域网的设备等),也可以是虚拟的软件平台(例如单体应用开发软件或在线平台等)。云端服务器用于存储应用的应用数据,并且允许具有权限的前端设备调用应用数据。

[0030] 移动端应用可以是软件(例如QQ、杀毒软件或WORD等)、网站(例如百度或当当等)或者手机APP,等等。在目前的开发中,一般ajax为核心的JSON数据传输模式,ajax是一种用于创建快速动态网页的技术,通过在后台与服务器进行少量数据交换,ajax可以使网页实现异步更新。这意味着可以在不重新加载整个网页的情况下,对网页的某部分进行更新,与传统的技术相比,网页(不使用ajax)如果需要更新内容,必须重载整个网页页面,ajax具有明显的减少数据传输优势。JSON是一种轻量级的数据交换格式。它基于ECMAScript的一个子集。JSON采用完全独立于语言的文本格式,但是也使用了类似于C语言家族的习惯(包括C、C++、C#、Java、JavaScript、Perl及Python等)。这些特性使JSON成为理想的数据交换语言。易于阅读和编写,同时也易于机器解析和生成。因此,由于移动端应用开发基本都采用ajax为核心的JSON数据传输,那么数据的类型与数据传输的方式有很大的通用性,例如,对数据库中的应用数据进行增、删、改及查等通用化功能来说,其逻辑都是一致的,而后端工程师在每开发一个应用时,这些通用化功能都需要进行手动书写代码来实现逻辑,这就使得后端工程师花费了大量时间做重复的事,浪费了人力和时间,对应用开发效率造成影响。

[0031] 为了解决上述问题,下面通过实施例对应用数据调用方法进行详细说明。

[0032] 请参阅图3,本发明实施例提供一种应用数据调用方法,包括:

[0033] 301、获取至少一个应用的应用数据;

[0034] 本实施例中,应用开发人员在应用开发平台上进行移动端应用开发过程中,获取到应用的应用数据,由于并未限制开发应用的数量,因此,应用的数量至少为一个。

[0035] 302、根据预置服务器部署规则和应用数据生成通用数据库,并将通用数据库部署至云端服务器;

[0036] 本实施例中,如果应用的开发已经完成,需要上线提供给用户使用,那么就需要将应用数据保存在云端服务器,以支持前端设备的用户通过云端服务器进行应用数据的传输,而将应用数据保存在云端服务器,需要以数据库的形式,根据预置服务器部署规则和应用数据生成通用数据库,将通用数据库部署至云端服务器,预置服务器部署规则是用户预先设置好的。

[0037] 303、获取通用化逻辑编辑指令及通用化逻辑数据;

[0038] 本实施例中,根据实际情况,数据传输无非增、删、改及查等等,因此应用的一些功能是通用性的,应用的通用化功能一般都是对云端服务器中的应用数据的增加、删减、修改、查看及验证等等。而通用化逻辑实际上就是处理应用数据的逻辑。例如,用户注册用户名,将用户名存入数据库的时候,用户名的字段应该是前台的用户名加上当前日期拼成的字符串,这就是应用数据的增加这一功能的逻辑。在应用开发的过程中,但是绝大多数的应用都有一部分的应用逻辑是相同的,称为通用化逻辑,为了改变现有的后端工程师手动书

写代码处理通用化逻辑的现状,采用通用化逻辑编辑指令来处理通用化逻辑,不需要后端工程师手动书写代码了。获取通用化逻辑编辑指令及通用化逻辑数据,通用化逻辑编辑指令用于编辑通用化逻辑数据,通用化逻辑数据用于表示处理应用数据的逻辑数据。

[0039] 304、根据通用化逻辑编辑指令和通用化逻辑数据生成服务器应用接口。

[0040] 本实施例中,获取到通用化逻辑编辑指令及通用化逻辑数据之后,根据通用化逻辑编辑指令和通用化逻辑数据生成服务器应用接口,前端设备通过服务器应用接口可以调用云端服务器中的应用数据,服务器应用接口包括用户注册接口、增加接口、删减接口、修改接口、查看接口及验证接口等等,需要说明的是,在实际应用中,还可能存在其他接口存在,以上只是举例进行说明,具体不做限定。

[0041] 本发明实施例中,应用开发系统获取至少一个应用的应用数据;根据预置服务器部署规则和应用数据生成通用数据库,并将通用数据库部署至云端服务器;获取通用化逻辑编辑指令及通用化逻辑数据,通用化逻辑编辑指令用于编辑通用化逻辑数据,通用化逻辑数据用于表示处理应用数据的逻辑数据;根据通用化逻辑编辑指令和通用化逻辑数据生成服务器应用接口,服务器应用接口用于调用云端服务器中的应用数据。由于应用数据库是根据预置服务器部署规则部署到云端服务器的,并且通用化逻辑处理是通过通用化逻辑编辑指令实现,前端设备通过生成的服务器应用接口就能进行调用应用数据,与现有技术相比,通用化逻辑不需要后端工程师对每一个通用化功能的服务器数据调用进行配置以及对通用化逻辑进行处理,因此,节省了人力和时间的耗费,提高了开发效率。

[0042] 上述图3所示的实施例的步骤304中根据通用化逻辑编辑指令和通用化逻辑数据生成服务器应用接口,还需要对数据进行处理,因为服务器应用接口是封装形式,具体如下:

[0043] 可选的,本发明的一些实施例中,根据通用化逻辑编辑指令和通用化逻辑数据生成服务器应用接口,包括:

[0044] 根据通用化逻辑编辑指令编辑通用化逻辑数据,得到逻辑字段,逻辑字段的字段类型为字符、文件或图片;

[0045] 对逻辑字段进行封装处理,生成服务器应用接口。

[0046] 本发明实施例中,通用化逻辑编辑指令是用于编辑通用化逻辑数据的,通用化逻辑数据的数据类型是代码的形式,为了简化通用化逻辑数据,并且提高交互性,那么根据通用化逻辑编辑指令可以编辑通用化逻辑数据,得到逻辑字段,逻辑字段的字段类型可以是字符、文件或者图片等。得到逻辑字段之后,对字段进行封装处理就能得到服务器应用接口,结合图4进行具体说明:

[0047] 401、逻辑字段;

[0048] 逻辑字段的字段类型为字符、文件或者图片等。

[0049] 402、语言转换;

[0050] 对逻辑字段的字段类型进行语言转换,例如可以转换为超文本预处理器(Hypertext Preprocessor,PHP)、JAVA或者node.js等等,PHP是一种通用开源脚本语言,语法吸收了C语言、Java和Perl的特点,利于学习,使用广泛,主要适用于Web开发领域。它可以比CGI或者Perl更快速地执行动态网页。用PHP做出的动态页面与其他的编程语言相比,PHP是将程序嵌入到HTML(标准通用标记语言下的一个应用)文档中去执行,执行效率比完全生

成HTML标记的CGI要高许多,PHP还可以执行编译后代码,编译可以达到加密和优化代码运行,使代码运行更快,因此,PHP非常适用于移动端应用开发。Java是一门面向对象编程语言,不仅吸收了C++语言的各种优点,还摒弃了C++里难以理解的多继承、指针等概念,因此Java语言具有功能强大和简单易用两个特征,在移动端应用开发中也是很好的选择。Node.js是一个基于Chrome JavaScript运行时建立的平台,用于方便地搭建响应速度快、易于扩展的网络应用。Node.js使用事件驱动,非阻塞I/O模型而得以轻量 and 高效,非常适合在分布式设备上运行数据密集型的实时应用,可知将逻辑字段转化为nodejs也是一种选择。需要说明的是,在实际应用或者以后的发展中,可能还会有更多适用的语言出现,上述三种只是举例说明,具体转化为哪种语言不做限定。

[0051] 403、封装处理;

[0052] 将语言转换后的逻辑字段进行封装处理。

[0053] 404、服务器应用接口。

[0054] 封装处理之后划分得到服务器应用接口,服务器应用接口包括增加接口、删减接口、修改接口、查看接口及验证接口等等。

[0055] 在上述实施例中,所有的逻辑字段都可以生成服务器应用接口,但是如果某个应用的应用数据库是不可读也不可写入应用数据,那么服务器应用接口是不允许生成的,那么就应该加密或者设置权限等。具体如下:

[0056] 可选的,本发明的一些实施例中,根据通用化逻辑编辑指令编辑通用化逻辑数据,得到逻辑字段之后,还包括:

[0057] 根据预置加密规则对逻辑字段进行加密,得到密钥信息;

[0058] 根据预置配置规则设置逻辑字段的访问权限信息;

[0059] 根据逻辑字段、密钥信息及访问权限信息生成逻辑数据表。

[0060] 本发明实施例中,预置加密规则为根据不可逆MD5算法在逻辑字段中添加特定字符或字符串来进行加密,并且得到token密钥,想要访问该逻辑字段从而生成服务器应用接口,就需要通过token密钥验证。预置配置规则为设置逻辑字段的访问权限,可视化配置哪些逻辑字段只能登陆用户查看,哪些字段只需要token密钥验证通过即可查看,根据逻辑字段、密钥信息及访问权限信息生成逻辑数据表,如下表1所示,

[0061] 表1

[0062]

逻辑字段	密钥信息	访问权限信息
逻辑字段1	Aabb	2
逻辑字段2	Babb	5
逻辑字段3	Cacc	3

[0063] 逻辑字段1的密钥信息为“Aabb”,访问权限信息为“2”,表示用户的等级为2以上可以访问;逻辑字段2的密钥信息为“Babb”,访问权限信息为“5”,表示用户的等级为5以上可以访问;逻辑字段3的密钥信息为“Cacc”,访问权限信息为“3”,表示用户的等级为3以上可以访问,

[0064] 可选的,本发明的一些实施例中,获取通用化逻辑编辑指令及通用化逻辑数据,包括:

- [0065] 接收用户根据通用化逻辑输入的通用化逻辑编辑指令；
- [0066] 根据通用化逻辑编辑指令确定对应的通用化逻辑；
- [0067] 获取通用化逻辑的通用化逻辑数据。
- [0068] 本发明实施例中，通用化逻辑编辑指令是由用户输入的，并且是根据通用化逻辑输入的，这样就给了开发人员可以删选通用化逻辑的条件。
- [0069] 可选的，本发明的一些实施例中，根据预置服务器部署规则和应用数据生成通用数据库，并将通用数据库部署至云端服务器，包括：
- [0070] 根据预置服务器部署规则从应用中确定通用化应用；
- [0071] 从应用数据中选择出通用化应用的应用数据，并生成通用数据库，通用数据库包括用户权限表、可查询日志文件及连接秘钥；
- [0072] 根据预置服务器部署规则将通用数据库部署至云端服务器。
- [0073] 本发明实施例中，服务器部署规则是预先设置好的，即预先设置好通用数据库是如何生成的，并且设置好通用数据库的服务器部署，并且通用数据库在云端服务器中的数据库名称为应用名称加上前缀，通用数据库存在于云端服务器上，并自动配置可以查阅的日志文件，通过初始密码加密通用数据库连接秘钥。使得前端设备不能随意访问通用数据库，在访问之前需要先根据用户权限表查看访问者是否达到访问权限，达到访问权限后，还需要根据连接秘钥与云端服务器中的通用数据库连接，并且只能访问可查询日志文件。
- [0074] 需要说明的是，应用开发系统还用于创建应用，具体为：
- [0075] 开发人员在应用开发系统的应用交互界面填入应用名称；
- [0076] 应用开发系统接收应用名称数据；
- [0077] 根据应用名称数据创建对应的数据库，该数据库的数据库名称为应用名称加上前缀；
- [0078] 当开发人员将所有的应用数据都输入后，即表示应用开发过程结束，应用的创建完成。
- [0079] 需要说明的是，在应用开发系统生成了服务器应用接口之后，前端设备还需要对服务器应用接口进行整理，具体的：
- [0080] 前端设备通过ajax或者无后端应用程序编程接口 (Application Programming Interface, API) 的js来对通用化逻辑进行组织；
- [0081] 直接从应用开发系统复制出服务器应用接口，对服务器应用接口进行修改串联整理。
- [0082] 上述实施例中介绍了本发明的应用数据调用方法，下面通过实施例对应用开发系统进行详细说明。
- [0083] 请参阅图5，本发明实施例提供一种应用开发系统，包括：
- [0084] 获取模块501，用于获取至少一个应用的应用数据；
- [0085] 详情请参考步骤301。
- [0086] 部署模块502，用于根据预置服务器部署规则和应用数据生成通用数据库，并将通用数据库部署至云端服务器；
- [0087] 详情请参考步骤302。
- [0088] 获取模块501，还用于获取通用化逻辑编辑指令及通用化逻辑数据，通用化逻辑编

辑指令用于编辑通用化逻辑数据,通用化逻辑数据用于表示处理应用数据的逻辑数据;

[0089] 详情请参考步骤303。

[0090] 接口生成模块503,用于根据通用化逻辑编辑指令和通用化逻辑数据生成服务器应用接口,服务器应用接口用于调用云端服务器中的应用数据。

[0091] 详情请参考步骤301。

[0092] 本发明实施例中,获取模块501获取至少一个应用的应用数据,部署模块502根据预置服务器部署规则和应用数据生成通用数据库,并将通用数据库部署至云端服务器,获取模块501获取通用化逻辑编辑指令及通用化逻辑数据,通用化逻辑编辑指令用于编辑通用化逻辑数据,通用化逻辑数据用于表示处理应用数据的逻辑数据,接口生成模块503根据通用化逻辑编辑指令和通用化逻辑数据生成服务器应用接口,服务器应用接口用于调用云端服务器中的应用数据。由于应用数据库是根据预置服务器部署规则部署到云端服务器的,并且通用化逻辑处理是通过通用化逻辑编辑指令实现,前端设备通过生成的服务器应用接口就能进行调用应用数据,与现有技术相比,通用化逻辑不需要后端工程师对每一个通用化功能的服务器数据调用进行配置以及对通用化逻辑进行处理,因此,节省了人力和时间的耗费,提高了开发效率。

[0093] 可选的,本发明的一些实施例中,

[0094] 接口生成模块503,具体用于根据通用化逻辑编辑指令编辑通用化逻辑数据,得到逻辑字段,逻辑字段的字段类型为字符、文件或图片;

[0095] 接口生成模块503,还用于对逻辑字段进行封装处理,生成服务器应用接口。

[0096] 本发明实施例中,通用化逻辑编辑指令是用于编辑通用化逻辑数据的,通用化逻辑数据的数据类型是代码,为了简化通用化逻辑数据,并且提高交互性,那么接口生成模块503根据通用化逻辑编辑指令可以编辑通用化逻辑数据,得到逻辑字段,逻辑字段的字段类型可以是字符、文件或者图片等。得到逻辑字段之后,对字段进行封装处理就能得到服务器应用接口。

[0097] 可选的,如图6所示,本发明的一些实施例中,应用开发系统还包括:

[0098] 数据处理模块504,具体用于根据预置加密规则对逻辑字段进行加密,得到密钥信息;

[0099] 数据处理模块504,还用于根据预置配置规则设置逻辑字段的访问权限信息;

[0100] 数据处理模块504,还用于根据逻辑字段、密钥信息及访问权限信息生成逻辑数据表。

[0101] 本发明实施例中,预置加密规则为根据不可逆MD5算法在逻辑字段中添加特定字符或字符串来进行加密,并且得到密钥信息,想要访问该逻辑字段从而生成服务器应用接口,就需要通过密钥验证。预置配置规则为设置逻辑字段的访问权限,数据处理模块504根据逻辑字段、密钥信息及访问权限信息生成逻辑数据表。

[0102] 可选的,本发明的一些实施例中,

[0103] 获取模块501,还用于接收用户根据通用化逻辑输入的通用化逻辑编辑指令;

[0104] 获取模块501,还用于根据通用化逻辑编辑指令确定对应的通用化逻辑;

[0105] 获取模块501,还用于获取通用化逻辑的通用化逻辑数据。

[0106] 本发明实施例中,通用化逻辑编辑指令是由用户输入的,并且是根据通用化逻辑

输入的,这样就给了开发人员可以删选通用化逻辑的条件。

[0107] 可选的,本发明的一些实施例中,,

[0108] 部署模块502,具体用于根据预置服务器部署规则从应用中确定通用化应用;

[0109] 部署模块502,还用于从应用数据中选择出通用化应用的应用数据,并生成通用数据库,通用数据库包括用户权限表、可查询日志文件及连接秘钥;

[0110] 部署模块502,还用于根据预置服务器部署规则将通用数据库部署至云端服务器。

[0111] 本发明实施例中,部署模块502根据预置服务器部署规则从应用中确定通用化应用,从应用数据中选择出通用化应用的应用数据,并生成通用数据库,通用数据库包括用户权限表、可查询日志文件及连接秘钥,根据预置服务器部署规则将通用数据库部署至云端服务器服务器部署规则是预先设置好的,即预先设置好通用数据库是如何生成的,并且设置好通用数据库的服务器部署,并且通用数据库在云端服务器中的数据库名称为应用名称加上前缀,通用数据库存在于云端服务器上,并自动配置可以查阅的日志文件,通过初始密码加密通用数据库连接秘钥。使得前端设备不能随意访问通用数据库,在访问之前需要先根据用户权限表查看访问者是否达到访问权限,达到访问权限后,还需要根据连接秘钥与云端服务器强中的通用数据库连接,并且只能访问可查询日志文件。

[0112] 以上介绍了应用开发系统的模块化结构的实施例,下面以应用开发系统为服务器为例,对评估装置的实体装置进行说明。

[0113] 潜在目标对象的评估装置以服务器为例,如图7所示,该评估装置可因配置或性能不同而产生比较大的差异,可以包括一个或一个以上中央处理器(central processing units,CPU)722(例如,一个或一个以上处理器)和存储器732,一个或一个以上存储应用程序742或数据744的存储介质730(例如一个或一个以上海量存储设备)。其中,存储器732和存储介质730可以是短暂存储或持久存储。存储在存储介质730的程序可以包括一个或一个以上模块(图示没标出),每个模块可以包括对服务器中的一系列指令操作。更进一步地,CPU722可以设置为与存储介质730通信,在服务器上执行存储介质730中的一系列指令操作。

[0114] 评估装置还可以包括一个或一个以上电源710,一个或一个以上无线网络接口750,一个或一个以上输入输出接口758,和/或,一个或一个以上操作系统741,例如Windows Server™,Mac OS X™,Unix™,Linux™,FreeBSD™等等。

[0115] 请参阅图7,本发明实施例提供一种应用开发系统,包括:

[0116] 无线网络接口750、CPU722及存储器732,无线网络接口750、CPU722及存储器703之间通过总线互相连接,存储器732中存储有计算机指令,CPU722通过执行计算机指令,从而实现以下方法:

[0117] 获取至少一个应用的应用数据;

[0118] 根据预置服务器部署规则和应用数据生成通用数据库,并将通用数据库部署至云端服务器;

[0119] 获取通用化逻辑编辑指令及通用化逻辑数据,通用化逻辑编辑指令用于编辑通用化逻辑数据,通用化逻辑数据用于表示处理应用数据的逻辑数据;

[0120] 根据通用化逻辑编辑指令和通用化逻辑生成服务器应用接口,服务器应用接口用于调用云端服务器中的应用数据。

[0121] 所属领域的技术人员可以清楚地了解到,为描述的方便和简洁,上述描述的系统,设备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。

[0122] 在本申请所提供的几个实施例中,应该理解到,所揭露的系统,设备和方法,可以通过其它的方式实现。例如,以上所描述的设备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,设备或单元的间接耦合或通信连接,可以是电性,机械或其它的形式。

[0123] 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。

[0124] 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。

[0125] 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,可以存储在一个计算机可读取存储介质中。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)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)、随机存取存储器(RAM, Random Access Memory)、磁碟或者光盘等各种可以存储程序代码的介质。

[0126] 以上所述,以上实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述实施例对本发明进行了详细的说明,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发明各实施例技术方案的精神和范围。

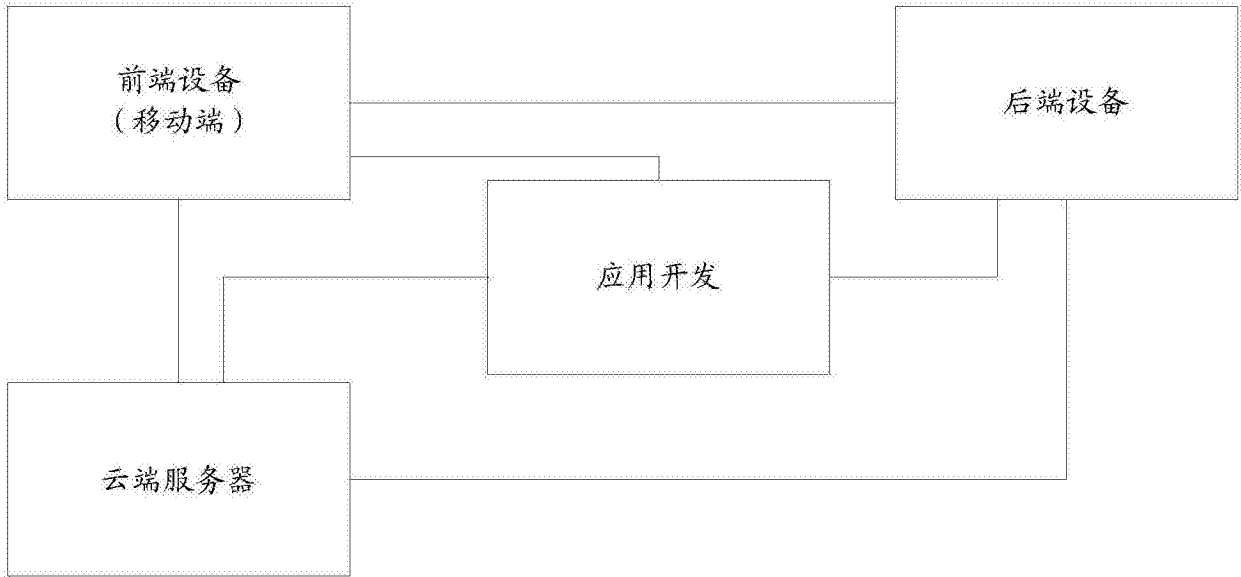


图1

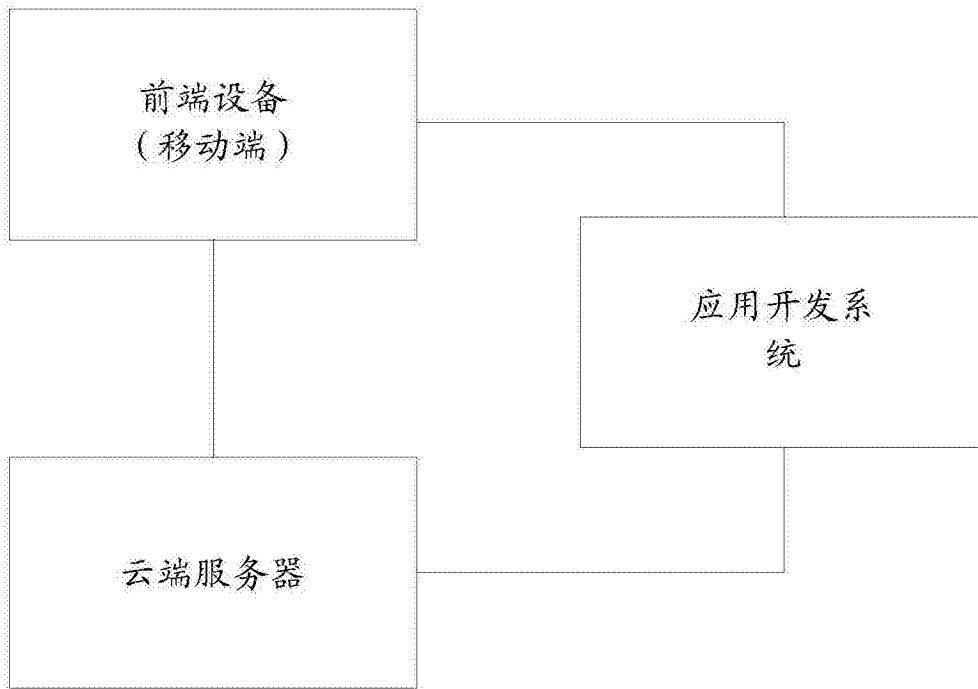


图2

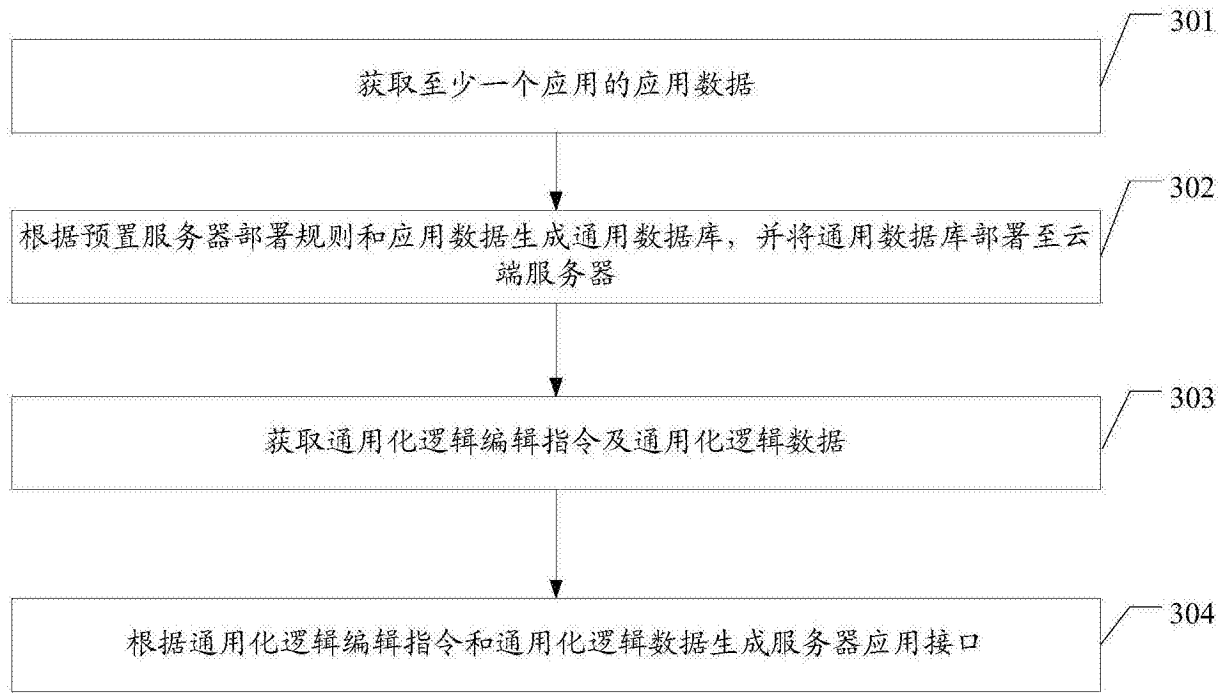


图3

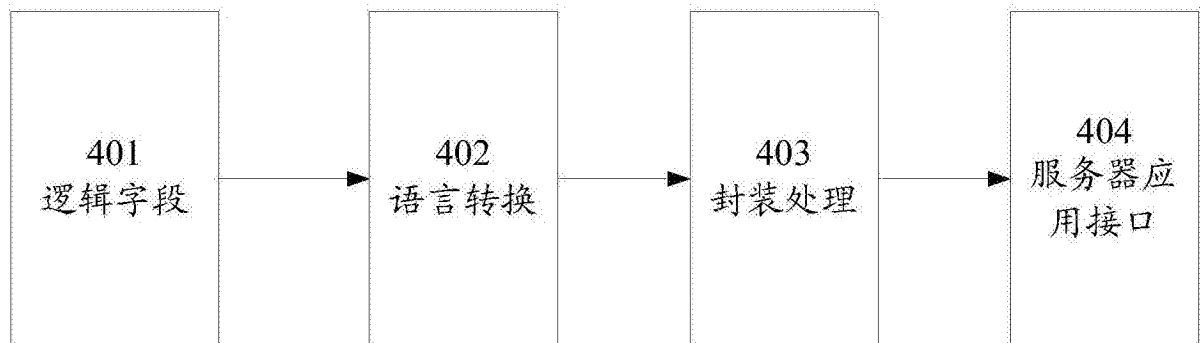


图4

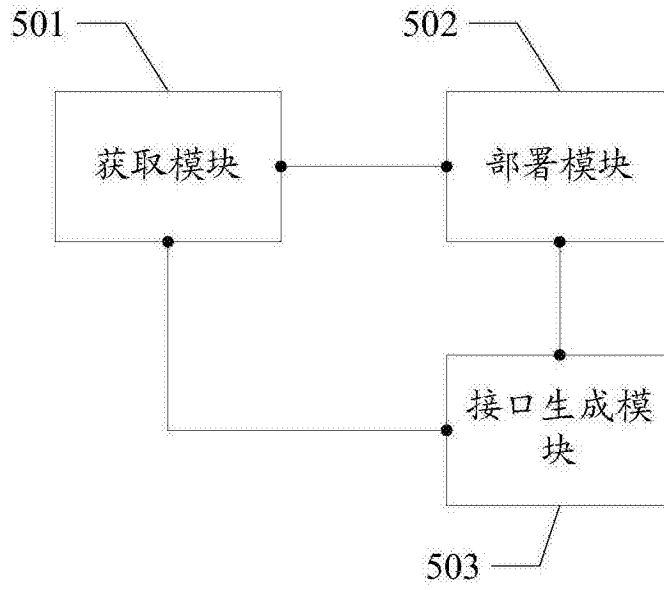


图5

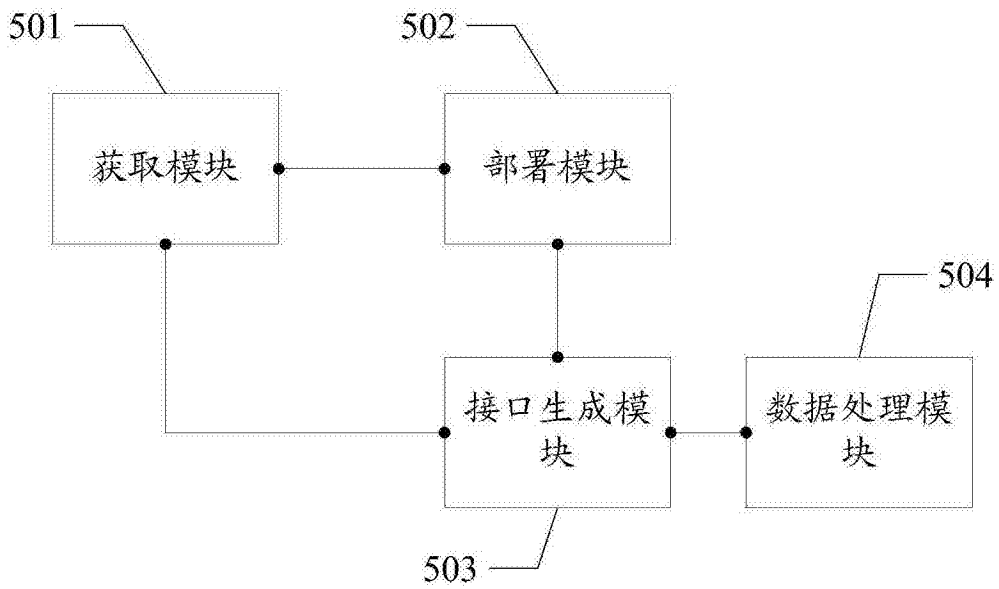


图6

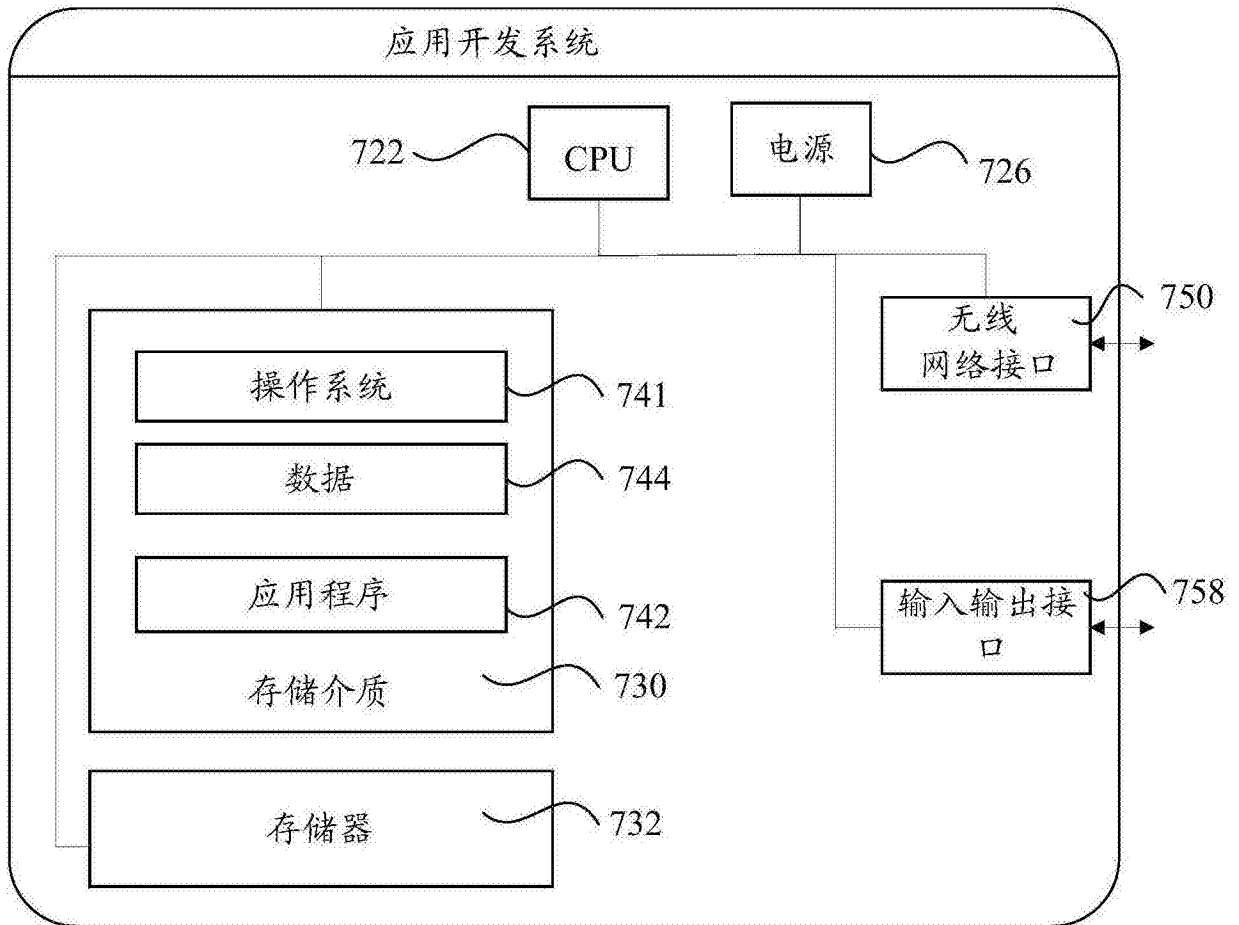


图7