



(12)发明专利申请

(10)申请公布号 CN 108076029 A

(43)申请公布日 2018.05.25

(21)申请号 201611032546.0

(22)申请日 2016.11.14

(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八卦岭
八卦三路平安大厦4楼

(72)发明人 徐光飞

(74)专利代理机构 深圳中一专利商标事务所
44237

代理人 张全文

(51) Int. Cl.

H04L 29/06(2006.01)

H04L 29/08(2006.01)

G06Q 10/06(2012.01)

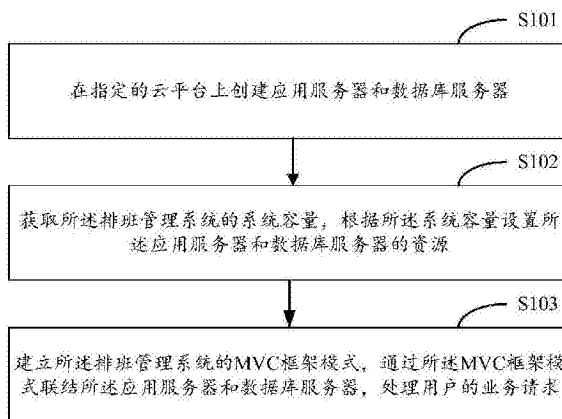
权利要求书2页 说明书6页 附图2页

(54)发明名称

排班管理系统的搭建方法及装置

(57)摘要

本发明适用于通信技术领域,提供了一种排班管理系统的搭建方法及装置,所述搭建方法包括:在指定的云平台上创建应用服务器和数据库服务器;获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源;建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求。本发明实现了Browser/Server结构的排班管理系统,将前端的页面展示和后端的业务逻辑分离,解决了现有Client/Server结构的排班管理系统难以扩展、无法支持SaaS模式下大量企业共同使用的问题。



1. 一种排班管理系统的搭建方法,其特征在于,所述搭建方法包括:
在指定的云平台上创建应用服务器和数据库服务器;
获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源;
建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求。
2. 如权利要求1所述的排班管理系统的搭建方法,其特征在于,在获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源之前,所述搭建方法还包括:
获取所述排班管理系统的性能参数,所述性能参数包括访问量、并发数、业务数据量、用户数量中的一种或者其任意组合;
根据所述性能参数估算所述排班管理系统的系统容量。
3. 如权利要求1所述的排班管理系统的搭建方法,其特征在于,所述通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求包括:
以所述应用服务器作为所述MVC框架模式中的控制器Controller,获取Browser端发送的业务请求;
判断所述业务请求是否合理,并在所述业务请求合理时,将所述业务请求发送至所述数据库服务器,以查询所述业务请求对应的数据资源;
获取所述数据库服务器返回的所述业务请求对应的数据资源,并将所述数据资源封装为业务模型Model;
将所述业务模型Model返回至所述Browser端,以使得所述Browser端在接收到所述业务模型Model后根据所述业务模型Model渲染页面。
4. 如权利要求1至3任一项所述的排班管理系统的搭建方法,其特征在于,所述应用服务器为中间件tomcat;
所述数据库服务器为PostgreSQL。
5. 一种排班管理系统的搭建装置,其特征在于,所述搭建装置包括:
创建模块,用于在指定的云平台上创建应用服务器和数据库服务器;
设置模块,用于获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源;
业务处理模块,用于建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求。
6. 如权利要求5所述的排班管理系统的搭建装置,其特征在于,所述搭建装置还包括:
获取模块,用于获取所述排班管理系统的性能参数,所述性能参数包括访问量、并发数、业务数据量、用户数量中的一种或者其任意组合;
估算模块,用于根据所述性能参数估算所述排班管理系统的系统容量。
7. 如权利要求5所述的排班管理系统的搭建装置,其特征在于,所述业务处理模块包括:
请求获取单元,用于以所述应用服务器作为所述MVC框架模式中的控制器Controller,获取Browser端发送的业务请求;

查询单元,用于判断所述业务请求是否合理,并在所述业务请求合理时,将所述业务请求发送至所述数据库服务器,以查询所述业务请求对应的数据资源;

封装单元,用于获取所述数据库服务器返回的所述业务请求对应的数据资源,并将所述数据资源封装为业务模型Model;

发送单元,用于将所述业务模型Model返回至所述Browser端,以使得所述Browser端在接收到所述业务模型Model后根据所述业务模型Model渲染页面。

8.如权利要求5至7任一项所述的排班管理系统的搭建装置,其特征在于,所述应用服务器为中间件tomcat;

所述数据库服务器为PostgreSQL。

排班管理系统的搭建方法及装置

技术领域

[0001] 本发明属于通信技术领域,尤其涉及一种排班管理系统的搭建方法及装置。

背景技术

[0002] 现有的排班管理产品为Client/Server结构,即基于中心化的服务器和客户端模式。Client/Server结构可以充分利用两端硬件环境的优势,将任务合理分配到Client端和Sever端来实现,降低了系统的通讯开销。然而,Client/Server结构中Client端的更新和操作繁琐,导致现有的排班管理系统难以扩展,无法支持SaaS(Software-as-a-Service,软件即服务)模式下大量企业共同使用的需求。

发明内容

[0003] 鉴于此,本发明实施例提供了一种排班管理系统的搭建方法及装置,以搭建Browser/Server结构的排班管理系统,提升系统的扩展性以及实现支持大量企业共同使用。

[0004] 第一方面,提供了一种排班管理系统的搭建方法,所述搭建方法包括:

[0005] 在指定的云平台上创建应用服务器和数据库服务器;

[0006] 获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源;

[0007] 建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求。

[0008] 进一步地,在获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源之前,所述搭建方法还包括;

[0009] 获取所述排班管理系统的性能参数,所述性能参数包括访问量、并发数、业务数据量、用户数量中的一种或者其任意组合;

[0010] 根据所述性能参数估算所述排班管理系统的系统容量。

[0011] 进一步地,所述通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求包括:

[0012] 以所述应用服务器作为所述MVC框架模式中的控制器Controller,获取Browser端发送的业务请求;

[0013] 判断所述业务请求是否合理,并在所述业务请求合理时,将所述业务请求发送至所述数据库服务器,以查询所述业务请求对应的数据资源;

[0014] 获取所述数据库服务器返回的所述业务请求对应的数据资源,并将所述数据资源封装为业务模型Model;

[0015] 将所述业务模型Model返回至所述Browser端,以使得所述Browser端在接收到所述业务模型Model后根据所述业务模型Model渲染页面。

[0016] 进一步地,所述应用服务器为中间件tomcat;

- [0017] 所述数据库服务器为PostgreSQL。
- [0018] 第二方面,提供了一种排班管理系统的搭建装置,所述搭建装置包括:
- [0019] 创建模块,用于在指定的云平台上创建应用服务器和数据库服务器;
- [0020] 设置模块,用于获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源;
- [0021] 业务处理模块,用于建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求。
- [0022] 进一步地,所述搭建装置还包括:
- [0023] 获取模块,用于获取所述排班管理系统的性能参数,所述性能参数包括访问量、并发数、业务数据量、用户数量中的一种或者其任意组合;
- [0024] 估算模块,用于根据所述性能参数估算所述排班管理系统的系统容量。
- [0025] 进一步地,所述业务处理模块包括:
- [0026] 请求获取单元,用于以所述应用服务器作为所述MVC框架模式中的控制器Controller,获取Browser端发送的业务请求;
- [0027] 查询单元,用于判断所述业务请求是否合理,并在所述业务请求合理时,将所述业务请求发送至所述数据库服务器,以查询所述业务请求对应的数据资源;
- [0028] 封装单元,用于获取所述数据库服务器返回的所述业务请求对应的数据资源,并将所述数据资源封装为业务模型Model;
- [0029] 发送单元,用于将所述业务模型Model返回至所述Browser端,以使得所述Browser端在接收到所述业务模型Model后根据所述业务模型Model渲染页面。
- [0030] 进一步地,所述应用服务器为中间件tomcat;
- [0031] 所述数据库服务器为PostgreSQL。
- [0032] 与现有技术相比,本发明实施例通过在指定的云平台上创建应用服务器和数据库服务器,并根据所述排班管理系统的系统容量,设置所述应用服务器和数据库服务器的资源;然后建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求;从而实现了Browser/Server结构的排班管理系统,能够支持大量企业的共同使用,并且所述排班管理系统通过将前端的页面展示和后端的业务逻辑分离,大大地提升了扩展性,有效地减轻了系统维护和升级的成本。

附图说明

- [0033] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例或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他附图。
- [0034] 图1是本发明实施例提供的排班管理系统的搭建方法的实现流程图;
- [0035] 图2是本发明实施例提供的排班管理系统的搭建方法中步骤S102的具体实现流程图;
- [0036] 图3是本发明实施例提供的排班管理系统的搭建装置的组成结构图。

具体实施方式

[0037] 为了使本发明的目的、技术方案及优点更加清楚明白,以下结合附图及实施例,对本发明进行进一步详细说明。应当理解,此处所描述的具体实施例仅仅用以解释本发明,并不用于限定本发明。

[0038] 本发明实施例通过在指定的云平台上创建应用服务器和数据库服务器,并根据所述排班管理系统的系统容量,设置所述应用服务器和数据库服务器的资源;然后建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求;从而实现了Browser/Server结构的排班管理系统,能够支持大量企业的共同使用,并且所述排班管理系统通过将前端的页面展示和后端的业务逻辑分离,大大地提升了扩展性,有效地减轻了系统维护和升级的成本。本发明实施例还提供了排班管理系统的搭建装置,以下分别进行详细的说明。

[0039] 图1是本发明实施例提供的排班管理系统的搭建方法的实现流程。

[0040] 在本发明实施例中,所述排班管理系统的搭建方法应用于终端设备上,所述终端设备包括但不限于计算机、服务器等。参阅图1,所述排班管理系统的搭建方法包括:

[0041] 在步骤S101中,在指定的云平台上创建应用服务器和数据库服务器。

[0042] 在这里,所述指定的云平台为Browser/Server结构的排班管理系统中Server端所部署的云平台。示例性地,当所述排班管理系统可以部署在平安云上时,即平安科技提供的平安云计算平台。本发明实施例在所述指定的云平台上创建Server端的应用服务器和数据库服务器。其中,所述应用服务器优选为中间件tomcat,所述数据库服务器优选为PostgreSQL数据库。本发明实施例通过结合PostgreSQL和中间件tomcat,实现了Server端的动态扩展框架。

[0043] 在步骤S102中,获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源。

[0044] 在创建应用服务器和数据库服务器之后,则根据所述排班管理系统的容量需求,设置所述应用服务器和数据库服务器的资源。其中,设置所述应用服务器和数据库服务器的资源具体包括:为所述应用服务器和数据库服务器分别配置相应的CPU和内存容量。

[0045] 进一步地,在获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源之前,所述搭建方法还可以包括:

[0046] 获取所述排班管理系统的性能参数;

[0047] 根据所述性能参数估算所述排班管理系统的系统容量。

[0048] 在这里,所述性能参数包括但不限于访问量、并发数、业务数据量、用户数量中的一种或者其任意组合。本发明实施例基于排班管理系统的性能参数来估算其对应的系统容量,并基于所述系统容量配置所述应用服务器和数据库服务器,进一步提高了排班管理系统的稳定性和扩展性。

[0049] 在步骤S103中,建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求。

[0050] 在这里,所述MVC是Model View Control,即模型Model-页面View-控制器Control的缩写。本发明实施例通过建立所述排班管理系统的MVC框架模式,其中,页面View的功能

在Browser端上实现,控制器Controller的功能则在Server端上的应用服务器实现,以使得用户通过Browser端访问所述排班管理系统时,Browser端的页面展示和Server端的业务逻辑是分开的,从而达到最大化的分层设计,使所述排班管理系统具备最大的扩展性。

[0051] 进一步地,图2示出了本发明实施例提供的步骤S103的具体实现流程。参阅图2,所述步骤S103通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求包括:

[0052] 在步骤S201中,以所述应用服务器作为所述MVC框架模式中的控制器Controller,获取Browser端发送的业务请求。

[0053] 在这里,本发明实施例以应用服务器作为MVC框架模式中的控制器Controller,以Browser端作为所述MVC框架模式中的页面View。用户通过智能终端(比如智能手机、平板电脑、计算机等)打开Browser访问所述排班管理系统时,Browser端首先加载html静态页面,然后加载angular js,以获取用户的业务请求,然后所述angular js通过ajax则将所述业务请求发送至所述MVC框架模式中的控制器Controller,即所述应用服务器。

[0054] 在步骤S202中,判断所述业务请求是否合理,并在所述业务请求合理时,将所述业务请求发送至所述数据库服务器,以查询所述业务请求对应的数据资源。

[0055] 用户的业务请求到达所述控制器Controller之后,所述控制器Controller验证所述业务请求的合理性,比如业务请求是否正确、当前用户是否具有所述请求的操作权限等等。当所述业务请求合理时,继续将所述业务请求转发至数据库服务器,以访问所述数据库服务器获取所述业务请求对应的数据资源。所述数据资源即用户请求的数据信息。

[0056] 在步骤S203中,获取所述数据库服务器返回的所述业务请求对应的数据资源,并将所述数据资源封装为业务模型Model。

[0057] 在步骤S204中,将所述业务模型Model返回至所述Browser端,以使得所述Browser端在接收到所述业务模型Model后根据所述业务模型Model渲染页面。

[0058] 在本发明实施例中,所述数据库服务器会将所获取到的资源返回至所述控制器Controller,所述控制器Controller接收到所述数据资源之后,将其封装成Model,并返回至所述Browser端,即MVC框架模式中的页面View。所述Browser端接收到Model之后,通过angular js使用所述Model渲染整个Browser页面,从而完成用户的一个请求操作。可见,在本实施例中,业务逻辑主要在Server端完成,Browser端仅需实现页面展示,Server端的业务逻辑与Browser端的页面展示是相互分开的,方便了开发人员对所述排班管理系统进行扩展、更新,提升了系统的扩展性;且在网络环境不佳时,Browser页面也能够正常显示,不会因为网络环境不佳出现空白页面,进而提升了Browser端用户的体验感。

[0059] 综上所述,本发明实施例通过在指定的云平台上创建应用服务器和数据库服务器,并根据所述排班管理系统的系统容量,设置所述应用服务器和数据库服务器的资源;然后建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求;从而实现了Browser/Server结构的排班管理系统,能够支持大量企业的共同使用,并且所述排班管理系统通过将前端的页面展示和后端的业务逻辑分离,大大地提升了扩展性,有效地减轻了系统维护和升级的成本。

[0060] 图3示出了本发明实施例提供的排班管理系统的搭建装置的组成结构,为了便于说明,仅示出了与本发明实施例相关的部分。

[0061] 在本发明实施例中,所述排班管理系统的搭建装置用于实现上述图1或图2实施例中所述的排班管理系统的搭建方法,可以是内置于终端设备的软件单元、硬件单元或者软硬件结合的单元。所述终端设备包括但不限于计算机、服务器等。

[0062] 参阅图3,所述排班管理系统的搭建装置包括:

[0063] 创建模块31,用于在指定的云平台上创建应用服务器和数据库服务器;

[0064] 设置模块32,用于获取所述排班管理系统的系统容量,根据所述系统容量设置所述应用服务器和数据库服务器的资源;

[0065] 业务处理模块33,用于建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求。

[0066] 进一步地,所述应用服务器为中间件tomcat;所述数据库服务器为PostgreSQL。

[0067] 进一步地,所述搭建装置还包括:

[0068] 获取模块34,用于获取所述排班管理系统的性能参数,所述性能参数包括访问量、并发数、业务数据量、用户数量中的一种或者其任意组合;

[0069] 估算模块35,用于根据所述性能参数估算所述排班管理系统的系统容量。

[0070] 本发明实施例基于排班管理系统的性能参数来估算其对应的系统容量,并基于所述系统容量配置所述应用服务器和数据库服务器,进一步提高了排班管理系统的稳定性和扩展性。

[0071] 进一步地,所述业务处理模块33包括:

[0072] 请求获取单元331,用于以所述应用服务器作为所述MVC框架模式中的控制器Controller,获取Browser端发送的业务请求;

[0073] 查询单元332,用于判断所述业务请求是否合理,并在所述业务请求合理时,将所述业务请求发送至所述数据库服务器,以查询所述业务请求对应的数据资源;

[0074] 封装单元333,用于获取所述数据库服务器返回的所述业务请求对应的数据资源,并将所述数据资源封装为业务模型Model;

[0075] 发送单元334,用于将所述业务模型Model返回至所述Browser端,以使得所述Browser端在接收到所述业务模型Model后根据所述业务模型Model渲染页面。

[0076] 在本实施例中,业务逻辑主要在Server端完成,Browser端仅需实现页面展示,Server端的业务逻辑与Browser端的页面展示是相互分开的,方便了开发人员对所述排班管理系统进行扩展、更新,提升了系统的扩展性;且在网络环境不佳时,Browser页面也能够正常显示,不会因为网络环境不佳出现空白页面,进而提升了Browser端用户的体验感。

[0077] 需要说明的是,本发明实施例中的装置可以用于实现上述方法实施例中的全部技术方案,其各个功能模块的功能可以根据上述方法实施例中的方法具体实现,其具体实现过程可参照上述实例中的相关描述,此处不再赘述。

[0078] 综上所述,本发明实施例通过在指定的云平台上创建应用服务器和数据库服务器,并根据所述排班管理系统的系统容量,设置所述应用服务器和数据库服务器的资源;然后建立所述排班管理系统的MVC框架模式,通过所述MVC框架模式联结所述应用服务器和数据库服务器,处理用户的业务请求;从而实现了Browser/Server结构的排班管理系统,能够支持大量企业的共同使用,并且所述排班管理系统通过将前端的页面展示和后端的业务逻辑分离,大大地提升了扩展性,有效地减轻了系统维护和升级的成本。

[0079] 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、或者计算机软件和电子硬件的结合来实现。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。

[0080] 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。

[0081] 在本申请所提供的几个实施例中,应该理解到,所揭露的方法及装置,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、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,装置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。

[0082] 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。

[0083] 另外,在本发明各个实施例中的各功能单元、模块可以集成在一个处理单元中,也可以是各个单元、模块单独物理存在,也可以两个或两个以上单元、模块集成在一个单元中。

[0084] 所述功能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。

[0085] 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,可轻易想到变化或替换,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所述以权利要求的保护范围为准。

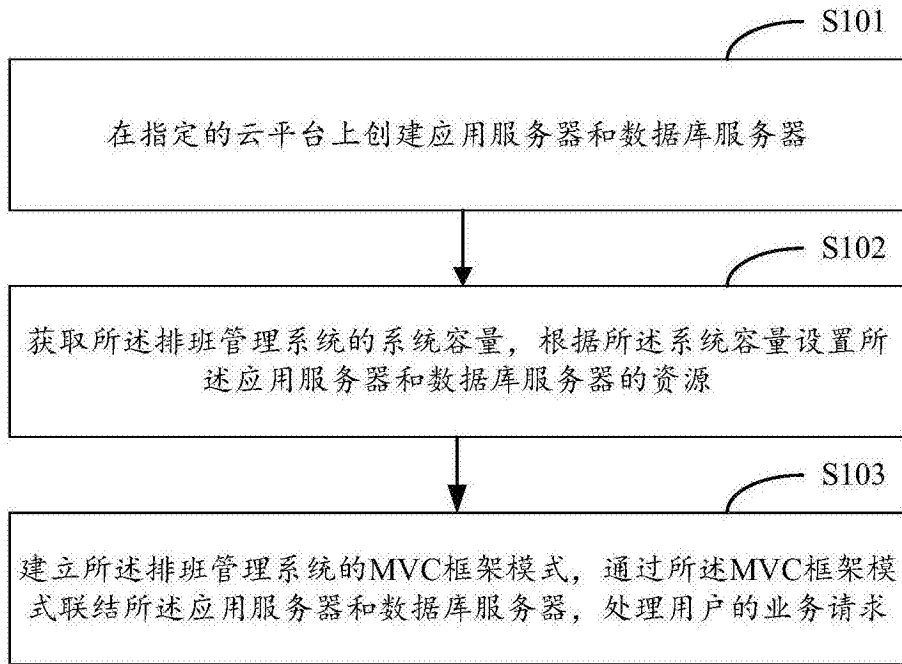


图1

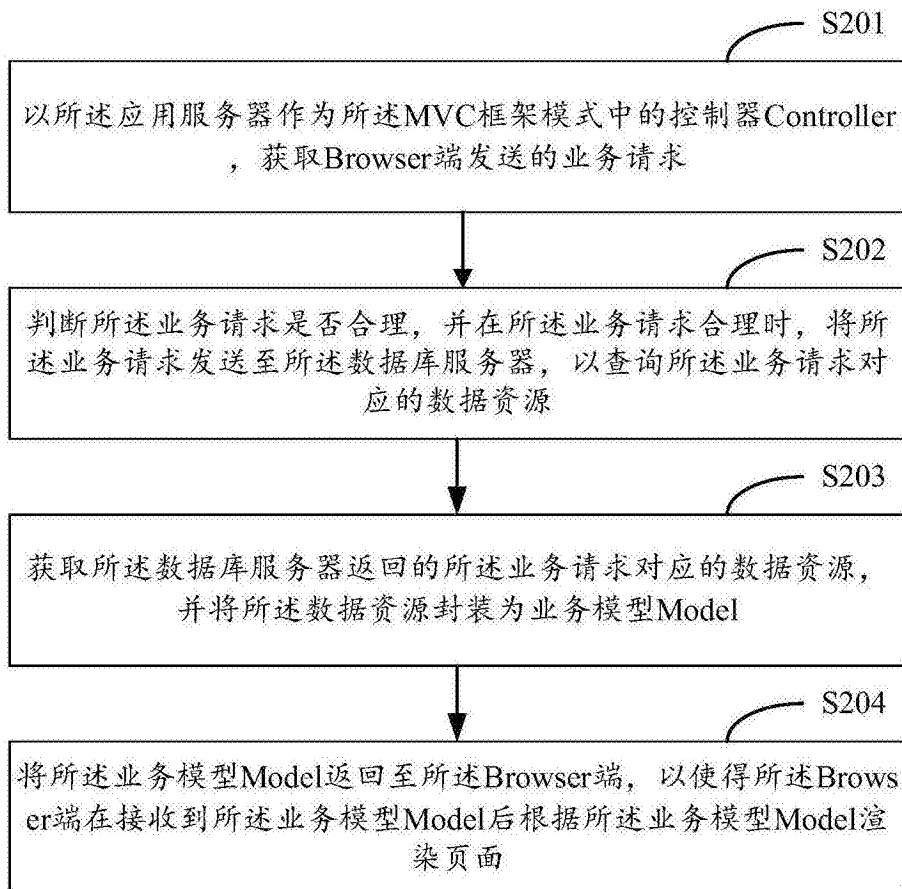


图2

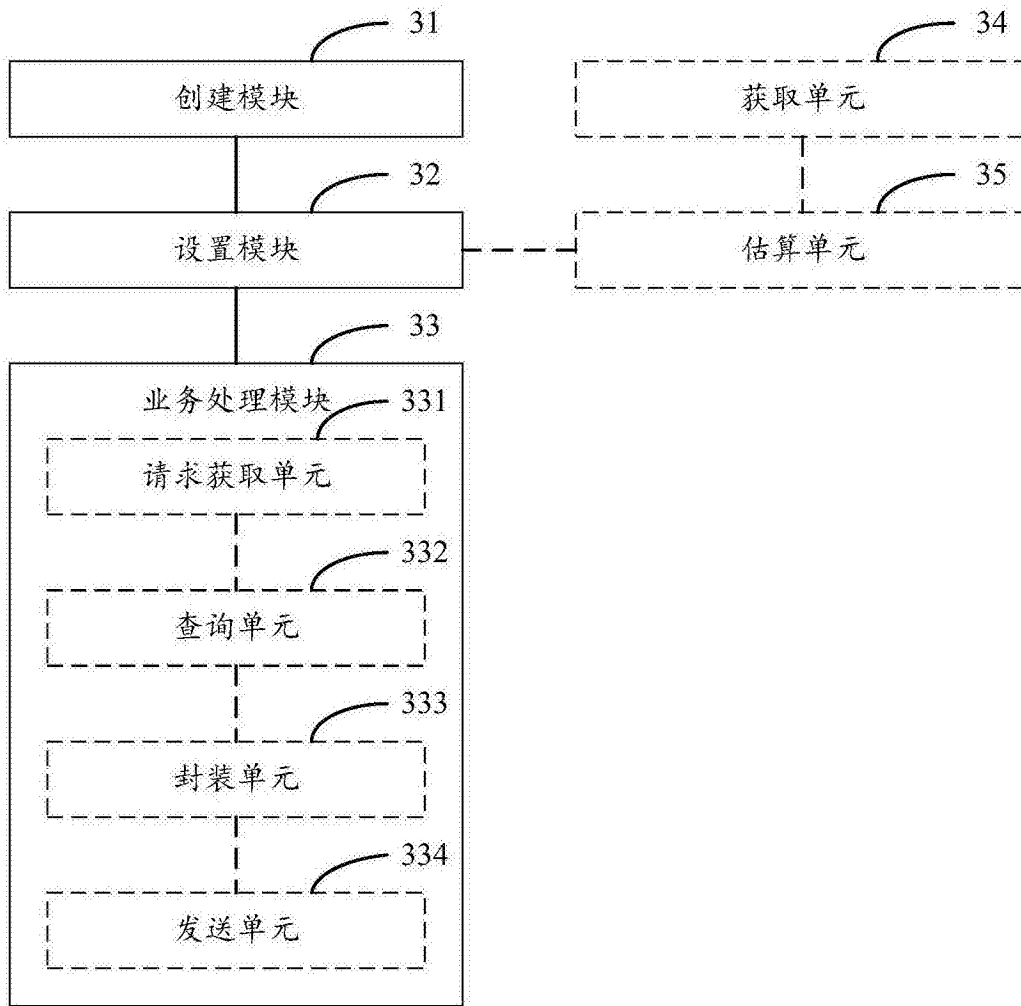


图3