



(21) 申请号 202410767733.1

G06Q 40/04 (2012.01)

(22) 申请日 2024.06.14

(56) 对比文件

(65) 同一申请的已公布的文献号

CN 113961458 A, 2022.01.21

申请公布号 CN 118350814 A

审查员 石慧君

(43) 申请公布日 2024.07.16

(73) 专利权人 北京宇信科技集团股份有限公司

地址 102627 北京市大兴区经济开发区科

苑路18号院3幢二层R2223室

(72) 发明人 李峰 肖姝 常振华 于浩

(74) 专利代理机构 北京卓岚智财知识产权代理

有限公司 11624

专利代理师 沈煜华

(51) Int. Cl.

G06Q 20/10 (2012.01)

G06Q 20/38 (2012.01)

权利要求书2页 说明书13页 附图7页

(54) 发明名称

分布式跨协议回调交易的生成方法、装置、
介质和设备

(57) 摘要

本发明提供一种分布式跨协议回调交易的生成方法、装置、介质和设备,属于金融科技领域,该方法包括:将接收到的交易请求分发至对应的业务微服务;根据交易报文中获取的业务交易配置信息确定回调交易的交易类型和协议类型,并根据交易类型对交易报文进行业务处理;根据业务处理结果和协议类型对回调接口进行交易动态配置以生成回调交易实例,并将回调交易实例注册到业务微服务的注册中心;根据回调交易实例对回调交易进行分组和版本匹配,根据匹配结果进行回调交易处理;根据回调交易处理的结果调用回调交易实例进行业务逻辑处理,对业务逻辑处理的结果和协议类型进行组装,完成回调交易生成和回调交易处理。本发明可与业务交易处理解耦,提升回调处理性能。



1. 一种分布式跨协议回调交易的生成方法,其特征在于,所述方法包括:

接收合作业务请求端发送的交易请求,根据合作业务请求端的接口信息将所述交易请求分发至对应的业务微服务;

根据所述交易请求从交易报文中获取业务交易配置信息,根据所述业务交易配置信息确定回调交易的交易类型和协议类型,并根据所述交易类型对所述交易报文进行业务处理,获得业务处理结果;

根据所述业务处理结果和所述协议类型对回调接口进行交易动态配置以生成回调交易实例,并将所述回调交易实例注册到业务微服务的注册中心;其中,所述的根据所述业务处理结果和所述协议类型对回调接口进行交易动态配置以生成回调交易实例,具体包括:根据业务处理结果进行协议转换,根据转换后的协议创建或使用已创建的交易回调消费者;所述交易回调消费者将交易结果的报文发送给所述业务微服务的交易回调生产者;所述交易回调生产者根据交易类型、目标交易名称、协议类型、交易报文和业务处理结果使用spring提供的类加载器生成回调交易实例;

根据所述回调交易实例对回调交易进行分组和版本匹配,获得匹配结果,并根据所述匹配结果进行回调交易处理,获得回调交易处理的结果;

根据所述回调交易处理的结果调用所述回调交易实例进行业务逻辑处理,对所述业务逻辑处理的结果和所述协议类型进行组装,完成回调交易生成和回调交易处理;其中,业务逻辑处理包括按照需求确定交易的分组信息和版本信息,按照协议类型配置回调交易是Dubbo协议还是http协议,以及配置交易所使用的ip地址、端口、服务名称和接口名称。

2. 根据权利要求1所述的方法,其特征在于,在所述的接收合作业务请求端发送的交易请求之前,还包括:

将业务交易之间的调用映射关系配置在数据库中;

从所述数据库中读取业务交易之间的调用映射关系;

根据所述调用映射关系将所述业务交易匹配到需要调用的回调交易。

3. 根据权利要求1所述的方法,其特征在于,在所述的根据所述交易类型对所述交易报文进行业务处理,获得业务处理结果之后,还包括:

对业务处理过程中的协议类型和API接口以配置表形式存储在数据库中。

4. 根据权利要求1所述的方法,其特征在于,所述的根据所述交易类型对所述交易报文进行业务处理,具体包括:

对所述交易请求中的交易报文进行校验,获得交易报文的校验结果;

根据所述交易报文的校验结果对校验后的交易报文按照所述交易类型进行业务处理,将所述业务处理的结果发送至关联系统;

接收所述关联系统的实时返回报文或异步查询结果,并将所述实时返回报文或异步查询结果回调至所述合作业务请求端。

5. 根据权利要求1所述的方法,其特征在于,所述回调交易处理具体包括:

判断是否为首次调用,若不是首次调用,则直接分发并调用回调交易实例,若是首次调用,则进行回调交易生成过程;

其中,所述回调交易生成过程具体包括:

使用监听管理获取业务微服务的相关配置信息;

根据所述相关配置信息通过注解的方式生成回调交易实例；

将所述回调交易实例通过业务微服务的动态注册的方式注册到注册中心。

6. 一种分布式跨协议回调交易的生成装置,其特征在於,所述装置包括:

交易请求接收和分发模块,用于接收合作业务请求端发送的交易请求,根据合作业务请求端的接口信息将所述交易请求分发至对应的业务微服务;

微服务处理模块,用于根据所述交易请求从交易报文中获取业务交易配置信息,根据所述业务交易配置信息确定回调交易的交易类型和协议类型,并根据所述交易类型对所述交易报文进行业务处理,获得业务处理结果;

动态配置模块,用于根据所述业务处理结果和所述协议类型对回调接口进行交易动态配置以生成回调交易实例,并将所述回调交易实例注册到业务微服务的注册中心;

匹配模块,用于根据所述回调交易实例对回调交易进行分组和版本匹配,获得匹配结果,并根据所述匹配结果进行回调交易处理,获得回调交易处理的结果;

回调交易生成模块,用于根据所述回调交易处理的结果调用所述回调交易实例进行业务逻辑处理,对所述业务逻辑处理的结果和所述协议类型进行组装,完成回调交易生成和回调交易处理。

7. 根据权利要求6所述的装置,其特征在於,所述装置还包括:

分发策略配置模块,用于将业务交易之间的调用映射关系配置在数据库中,从所述数据库中读取业务交易间的调用映射关系,根据所述调用映射关系将所述业务交易匹配到需要调用的回调交易。

8. 一种电子设备,其特征在於,包括:

一个或多个处理器;

存储装置,用于存储一个或多个程序,当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如权利要求1-5中任一项所述的分布式跨协议回调交易的生成方法。

9. 一种计算机可读介质,其上存储有计算机程序,其特征在於,该程序被处理器执行时实现如权利要求1-5中任一项所述的分布式跨协议回调交易的生成方法。

分布式跨协议回调交易的生成方法、装置、介质和设备

技术领域

[0001] 本发明涉及金融科技领域,尤其涉及一种分布式跨协议回调交易的生成方法、装置、介质和设备。

背景技术

[0002] 随着软件技术的飞速发展,交易系统功能的复杂性越来越高,性能要求越来越高和业务变化越来越快,业务的处理逻辑和客户端的标准不统一,众多行业的企业在建设信息系统时选择以分布式服务方式,若同一系统中存在多种分布式微服务,由于不同框架下业务应用交易处理无法统一,因此,如何快速高效的响应业务的变化且有效控制重复成本投入的开发面临着巨大的挑战。

[0003] 在系统迭代和业务运营过程中,需要根据业务运营场景对业务处理交易进行不断的迭代,因此公共服务中与关联系统交互的微服务,由于系统间协议、规范、交易标准等不一致,导致目前同一个API (Application Programming Interface,应用程序编程接口)网关无法处理不同的业务逻辑,同时无法兼容不同微服务。随着业务开展时间的增加,业务量和业务类型会随之持续增多,同时业务发展的变化,和行业对产品标准的管理日益增多,管理机构对业务的运营、监控、分析和数字化要求不断提高,传统方式的网关迭代优化开发不但成本大,质量很难保障,且响应需求的时间长,导致信息系统更新效率不高。

[0004] 现有业务信息系统,存在多种微服务架构如Dubbo、SpringCloud等,目前实现Dubbo接口与SpringCloud接口的交互有两种方式:一是通过兼容两种框架的注册和发现中间件,整合两类微服务;二是使用微服务转换配置工具,对SpringCloud接口进行配置。但是目前这两种方式,业务交易无法在不同协议下进行回调处理,需要按交易和协议的不同提供多个回调服务实例,占用过多系统资源,降低了系统性能,增大了信息系统的开发、测试、上线及业务验证投入。

发明内容

[0005] 有鉴于此,本发明实施例的目的在于提供一种分布式跨协议回调交易的生成方法、装置、介质和设备,以解决现有技术中存在的上述问题。

[0006] 为达到上述目的,第一方面,本发明实施例一种分布式跨协议回调交易的生成方法,所述方法包括:

[0007] 接收合作业务请求端发送的交易请求,根据合作业务请求端的接口信息将所述交易请求分发至对应的业务微服务;

[0008] 根据所述交易请求从交易报文中获取业务交易配置信息,根据所述业务交易配置信息确定回调交易的交易类型和协议类型,并根据所述交易类型对所述交易报文进行业务处理,获得业务处理结果;

[0009] 根据所述业务处理结果和所述协议类型对回调接口进行交易动态配置以生成回调交易实例,并将所述回调交易实例注册到业务微服务的注册中心;

[0010] 根据所述回调交易实例对回调交易进行分组和版本匹配,获得匹配结果,并根据所述匹配结果进行回调交易处理,获得回调交易处理的结果;

[0011] 根据所述回调交易处理的结果调用所述回调交易实例进行业务逻辑处理,对所述业务逻辑处理的结果和所述协议类型进行组装,完成回调交易生成和回调交易处理。

[0012] 在一些可能的实施方式中,在所述的接收合作业务请求端发送的交易请求之前,还包括:

[0013] 将业务交易之间的调用映射关系配置在数据库中;

[0014] 从所述数据库中读取业务交易之间的调用映射关系;

[0015] 根据所述调用映射关系将所述业务交易匹配到需要调用的回调交易。

[0016] 在一些可能的实施方式中,在所述的根据所述交易类型对所述交易报文进行业务处理,获得业务处理结果之后,还包括:

[0017] 对业务处理过程中的协议类型和API接口以配置表形式存储在数据库中。

[0018] 在一些可能的实施方式中,所述的根据所述交易类型对所述交易报文进行业务处理,具体包括:

[0019] 对所述交易请求中的交易报文进行校验,获得交易报文的校验结果;

[0020] 根据所述交易报文的校验结果对校验后的交易报文按照所述交易类型进行业务处理,将所述业务处理的结果发送至关联系统;

[0021] 接收所述关联系统的实时返回报文或异步查询结果,并将所述实时返回报文或异步查询结果回调至所述合作业务请求端。

[0022] 在一些可能的实施方式中,所述的根据所述业务处理结果和所述协议类型对回调接口进行交易动态配置以生成回调交易实例,具体包括:

[0023] 根据交易类型、目标交易名称、协议类型、交易报文和业务处理结果,使用spring提供的类加载器生成回调交易实例。

[0024] 在一些可能的实施方式中,所述回调交易处理具体包括:

[0025] 判断是否为首次调用,若不是首次调用,则直接分发并调用回调交易实例,若是首次调用,则进行回调交易生成过程;

[0026] 其中,所述回调交易生成过程具体包括:

[0027] 使用监听管理获取业务微服务的相关配置信息;

[0028] 根据所述相关配置信息通过注解的方式生成回调交易实例;

[0029] 将所述回调交易实例通过业务微服务的动态注册的方式注册到注册中心。

[0030] 第二方面,本发明实施例提供了一种分布式跨协议回调交易的生成装置,所述装置包括:

[0031] 交易请求接收和分发模块,用于接收合作业务请求端发送的交易请求,根据合作业务请求端的接口信息将所述交易请求分发至对应的业务微服务;

[0032] 微服务处理模块,用于根据所述交易请求从交易报文中获取业务交易配置信息,根据所述业务交易配置信息确定回调交易的交易类型和协议类型,并根据所述交易类型对所述交易报文进行业务处理,获得业务处理结果;

[0033] 动态配置模块,用于根据所述业务处理结果和所述协议类型对回调接口进行交易动态配置以生成回调交易实例,并将所述回调交易实例注册到业务微服务的注册中心;

[0034] 匹配模块,用于根据所述回调交易实例对回调交易进行分组和版本匹配,获得匹配结果,并根据所述匹配结果进行回调交易处理,获得回调交易处理的结果;

[0035] 回调交易生成模块,用于根据所述回调交易处理的结果调用所述回调交易实例进行业务逻辑处理,对所述业务逻辑处理的结果和所述协议类型进行组装,完成回调交易生成和回调交易处理。

[0036] 在一些可能的实施方式中,所述装置还包括:

[0037] 分发策略配置模块,用于将业务交易之间的调用映射关系配置在数据库中,从所述数据库中读取业务交易间的调用映射关系,根据所述调用映射关系将所述业务交易匹配到需要调用的回调交易。

[0038] 第三方面,本发明实施例提供了一种电子设备,包括:

[0039] 一个或多个处理器;

[0040] 存储装置,用于存储一个或多个程序,当所述一个或多个程序被所述一个或多个处理器执行,使得所述一个或多个处理器实现如第一方面任意一种所述的方法。

[0041] 第四方面,本发明实施例提供了一种计算机可读介质,其上存储有计算机程序,该程序被处理器执行时实现如第一方面任意一种所述的方法。

[0042] 上述技术方案具有如下有益效果:

[0043] 本发明提供了一种分布式跨协议回调交易的生成方法、装置、介质和设备该方法包括:合作业务请求端的接口信息将交易请求分发至对应的业务微服务;根据交易请求从交易报文中获取业务交易配置信息,根据业务交易配置信息确定回调交易的交易类型和协议类型,并根据交易类型对交易报文进行业务处理;根据业务处理结果和协议类型对回调接口进行交易动态配置以生成回调交易实例,并将回调交易实例注册到业务微服务的注册中心;根据回调交易实例对回调交易进行分组和版本匹配,根据匹配结果进行回调交易处理,获得回调交易处理的结果;根据回调交易处理的结果调用回调交易实例进行业务逻辑处理,对业务逻辑处理的结果和协议类型进行组装,完成回调交易生成和回调交易处理。

[0044] 本发明通过支持不同回调协议,且可以根据交易信息处理回调交易路由节点;消除客户端与服务处理端的依赖,降低信息系统建设及后期维护复杂度,根据业务交易配置信息及交易协议等,动态组装不同协议,实现跨交易类型、服务协议或中间件通用化处理业务交易的回调功能,并可以分发到不同微服务处理节点,提高交易处理系统的通用性和处理性能。

附图说明

[0045] 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,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的附图。

[0046] 图1是本发明实施例的一种分布式跨协议回调交易的生成方法的流程图;

[0047] 图2是本发明实施例的一种分布式跨协议回调交易的部署图;

[0048] 图3是本发明实施例的一种放款业务处理流程的部署图;

[0049] 图4是本发明实施例的一种放款申请业务处理的流程图;

- [0050] 图5是本发明实施例的一种放款申请结果查询业务处理的流程图；
- [0051] 图6是本发明实施例的一种放款申请结果查询业务处理的流程图；
- [0052] 图7是本发明实施例的一种分布式跨协议回调交易生成的流程图；
- [0053] 图8是本发明实施例的一种分布式跨协议回调交易的生成装置的结构框图；
- [0054] 图9是本发明实施例的另一种分布式跨协议回调交易的生成装置的结构框图；
- [0055] 图10是本发明实施例的一种电子设备的功能框图。

具体实施方式

[0056] 下面将参照附图更详细地描述本公开的实施例。虽然附图中显示了本公开的某些实施例,然而应当理解的是,本公开可以通过各种形式来实现,而且不应该被解释为限于这里阐述的实施例,相反提供这些实施例是为了更加透彻和完整地理解本公开。应当理解的是,本公开的附图及实施例仅用于示例性作用,并非用于限制本公开的保护范围。

[0057] 应当理解,本公开的方法实施方式中记载的各个步骤可以按照不同的顺序执行,和/或并行执行。此外,方法实施方式可以包括附加的步骤和/或省略执行示出的步骤。本公开的范围在此方面不受限制。

[0058] 本文使用的术语“包括”及其变形是开放性包括,即“包括但不限于”。术语“基于”是“至少部分地基于”。术语“一个实施例”表示“至少一个实施例”;术语“另一实施例”表示“至少一个另外的实施例”;术语“一些实施例”表示“至少一些实施例”。其他术语的相关定义将在下文描述中给出。

[0059] 需要注意,本公开中提及的“第一”、“第二”等概念仅用于对不同的装置、模块或单元进行区分,并非用于限定这些装置、模块或单元所执行的功能的顺序或者相互依存关系。

[0060] 需要注意,本公开中提及的“一个”、“多个”的修饰是示意性而非限制性的,本领域技术人员应当理解,除非在上下文另有明确指出,否则应该理解为“一个或多个”。

[0061] 本公开实施方式中的多个装置之间所交互的消息或者信息的名称仅用于说明性的目的,而并不是用于对这些消息或信息的范围进行限制。

[0062] 术语说明:

[0063] API网关:是一个软件工程中术语,本发明中的主要功能是在多个系统之间交易处理的中介者,根据业务交易处理的需要,处理不同API的调用。

[0064] Dubbo:是一个开源的高性能、轻量级的Java RPC(Remote Procedure Call,远程过程调用)框架,远程调用采用Dubbo协议,接口一般是Java的Service接口,格式固定。

[0065] SpringCloud:基于Spring Boot框架,为Java开发人员提供了一种快速构建分布式系统的方法,远程调用采用Http(HyperText Transfer Protocol,超文本传输协议)协议做远程调用,接口一般是Rest(Representational State Transfer,表述性状态转移)风格。

[0066] 图1是一种分布式跨协议回调交易的生成方法的流程图,图2是本发明实施例的一种分布式跨协议回调交易的部署图,如图1和图2所示,该方法包括:

[0067] 步骤S1,接收合作业务请求端发送的交易请求,根据合作业务请求端的接口信息将交易请求分发至对应的业务微服务。

[0068] 本实施例中,对多种API协议进行统一管理,通过API接口接收交易和回调交易均

按合作机构类型、产品编号、交易类型、交易流水等,对交易请求进行分发和处理。具体的,交易请求是指外部合作机构系统发起的接口调用,每个合作机构会有一个合作业务系统,API网关根据合作业务请求端的接口信息,例如ip、端口、URL(Uniform Resource Locator,统一资源定位符)等通过微服务端口将交易请求分配给对应的业务微服务。

[0069] 步骤S2,根据交易请求从交易报文中获取业务交易配置信息,根据业务交易配置信息确定回调交易的交易类型和协议类型,并根据交易类型对交易报文进行业务处理,获得业务处理结果。本实施例中,对接收和回调交易分别进行解析,从接收交易中获取业务交易配置信息,从回调交易中获取回调交易的交易类型和协议类型,并根据交易类型对交易报文通过业务处理1、业务处理2……业务处理n等进行业务处理,下面以放款为例加以详细说明。

[0070] 图3是本发明实施例的一种放款业务处理流程的部署图,如图3所示,本实施例中,交易类型可以为放款,授信、提额、还款和结清等,以放款为例,如图3所示,本实施例中,可以包括合作业务1、业务合作2……业务合作n等多个合作方,每一个业务合作对应一个微服务,例如业务1微服务、业务2微服务……业务n微服务,每一个微服务中部署有多个交易,例如交易1消费者、交易2消费者和交易3生产者,放款申请消费者、放款查询消费者和放款回调生产者作为放款业务的一组交易,交易消费者一般指发起交易请求的调用方,为客户端;交易生产者一般指提供交易处理的服务端,放款申请消费者为放款申请交易的调用方,回调生产者用于提供回调交易的服务端。

[0071] 具体的,业务微服务消费者对交易报文进行合法性和完整性校验,交易合法性是指报文是否符合接口约定,例如:报文长度、格式(文本、数值等)、必填项等;交易的完整性是指在交易过程中,报文的内容、结构和顺序保持完整,没有被未经授权的修改、破坏或丢失,检查报文后将交易请求发送给微服务交易生产者处理。业务交易生产者根据业务需求,处理放款交易前处理:业务交易生产者是根据需求对报文进行解析,从报文获取客户标识和申请额度信息,再根据客户标识,在信息系统中关联查询到授信状态、客户信息及合同信息。放款业务微服务根据放款类型及相关信息,将申请提交到关联系统,例如银行系统中的核心放款、超网放款、三方支付放款系统),其中放款类型指通过超网、三方支付或核心放款;相关信息包括:交易时间、营业日期、申请编号、申请金额等。根据关联系统的实时返回报文或异步查询结果,将处理结果信息返回给业务微服务,业务微服务最终将结果回调返回给合作业务系统。其中,实时返回报文指的是将放款是否成功,业务状态码等接口约定信息返回给请求方;但是,合作方不接收业务返回报文信息的,会按照请求交易的编号查询放款交易结果,即异步查询。

[0072] 步骤S3,根据业务处理结果和协议类型对回调接口进行交易动态配置以生成回调交易实例,并将回调交易实例注册到业务微服务的注册中心。

[0073] 本实施例中,对业务处理结果和协议类型进行组装,生成回调交易实例,以供API调用;具体的,本实施例根据交易信息来动态配置并生成回调交易实例,即根据业务处理结果和协议类型进行协议转换,并按配置转发队列,据业务交易的解析和分发配置,确定要调用目标交易的协议类型,使用spring类型加载器及注解管理等工具,创建http交易或者Dubbo交易,实现业务交易的协议转换。最终实现通用交易微服务的合理扩展和扩容,减少部署实例数量和版本发布次数;从而降低运维人员工作量,上线复杂度。

[0074] 本发明通过交易动态配置,可以支持不同回调协议,以及根据交易信息处理回调交易路由节点;消除客户端与服务处理端的依赖,降低信息系统建设及后期维护复杂度,根据业务交易配置信息及交易协议等,动态组装不同协议,实现跨交易类型、服务协议或中间件通用化处理业务交易的回调功能,并可以分发到不同微服务处理节点,提高交易处理系统的通用性和处理性能。

[0075] 步骤S4,根据回调交易实例对回调交易进行分组和版本匹配,获得匹配结果,并根据匹配结果进行回调交易处理,获得回调交易处理的结果。

[0076] 具体的,分布式微服务框架本身不支持不同分组和版本间调用;但是在一些场景下,需要不同分组的业务交易调用某一个固定分组的业务交易。并按照业务逻辑要求,将回调交易对应关系配置在分发配置管理器中。

[0077] 步骤S5,根据回调交易处理的结果调用回调交易实例进行业务逻辑处理,对业务逻辑处理的结果和协议类型进行组装,完成回调交易生成和回调交易处理。

[0078] 本实施例中,业务逻辑处理包括按照需求确定交易的分组和版本信息,按照协议类型配置回调交易是Dubbo协议还是http协议,以及配置交易要具体的使用的ip地址、端口、服务名称、接口名称等,根据这些协议类型及其配置和业务逻辑处理的获得的分组和版本信息进行组装后生成回调交易。

[0079] 本发明实施例将原本基于Dubbo协议转换为基于http协议的调用,以及根据交易配置动态生成Dubbo交易或http交易并完成与中间件的服务注册和发现,来降低业务交易设计难度,减少代码开发数量;从而降低开发人员工作量和难度,根据业务调整频率对应开发量情况,在微服务中选取应用扩展配置初始化、微服务生产者和消费者实例化、以及微服务注册及发现这几个过程,来设计和实现动态交易管理。

[0080] 在一些实施例中,在步骤S1中的接收合作业务请求端发送的交易请求之前,还可以包括:

[0081] 将业务交易之间的调用映射关系配置在数据库中,从数据库中读取业务交易间的调用映射关系,根据调用映射关系将业务交易匹配到需要调用的回调交易。

[0082] 本实施例中,不同协议间根据业务需求配置映射关系,保存在数据库中,若回调交易协议与请求交易不同,则取得交易协议;本实施例提供两个交易接口,一个http协议的,一个Dubbo协议的;配置信息包括交易类型,接口URL、分组、版本,保存在数据库中;通过维护配置信息,对支持的回调交易进行管理,若回调交易请求的信息在支持范围内,调用时对回调交易进行实例化生成。

[0083] 在一些实施例中,在步骤S2中的根据交易类型对交易报文进行业务处理,获得业务处理结果之后,还可以包括:对业务处理过程中的协议类型和API接口进行统一管理,并以配置表形式存储在数据库中。

[0084] 本实施例在业务处理的过程中,将业务处理涉及到的所有交易的通讯协议种类和API接口,进行统一标准管理,以配置表形式存储在数据库中。

[0085] 在一些实施例中,所述的根据所述交易类型对所述交易报文进行业务处理,具体包括:

[0086] 对交易请求中的交易报文进行校验,获得交易报文的校验结果;根据交易报文的校验结果对校验后的交易报文按照交易类型进行业务处理,将业务处理的结果发送至关联

系统;接收关联系统的实时返回报文或异步查询结果,并将实时返回报文或异步查询结果回调至合作业务请求端。

[0087] 作为一个举例说明,以放款为例,在业务处理放款过程中,放款业务处理对业务微服务提交的放款类型对报文进行转换后,将业务合作系统提交的交易报文发送至银行系统。本实施例中,放款申请生产者、放款查询生产者和放款回调消费者共同完成放款业务处理流程。通过放款申请生产者处理放款申请,放款查询生产者进行放款申请结果查询,放款回调消费者进行放款申请结果回调。

[0088] 图4是本发明实施例的一种放款申请业务处理的流程图,如图4所示,其包括如下步骤:

[0089] S11,接收合作业务提交的放款申请交易;

[0090] S12,检查放款申请交易中是否存在放款申请号;如果存在放款申请号,执行步骤S13,如果不存在放款申请号,执行步骤S15;

[0091] S13,判断是否为失败交易,若是失败交易,记录交易时间及交易数据,执行S16,若不是失败交易,执行S14;

[0092] S14,判断是否为超时交易状态,若未超时,则执行S17,若超时,则记录超时信息,执行S16;

[0093] S15,判断放款申请交易是否通过放款规则,如果放款申请交易通过放款规则,执行S17,如果放款申请交易没有通过放款规则,执行步骤S18。本实施例中,放款规则包括放款额度和信审状态等。

[0094] 本实施例中,放款额度为放款申请交易报文中客户最大放款金额,一般客户放款额度是金融机构根据客户的信用状况、借款用途、经济情况、抵押品或担保等多个因素综合评估后确定的。信审状态是指贷款信审状态,贷款信审状态指的是贷款申请在审核过程中的状态,一般有申请中、通过、不通过,已用信及贷款状态等。例如,张三申请一张信用卡时,会根据收入信息等,确定张三信用卡额度即为放款额度。张三信用卡申请未通过、已通过或者其他信用卡有使用或逾期即为信审状态。

[0095] S16,执行放款申请交易失败处理;

[0096] 本实施例中,交易状态参数记为失败标识,在日志文件中输出记录日志,将数据库对应的放款申请表对应的明细记录更新交易结果字段为失败。

[0097] S17,执行放款申请交易,记录交易状态和交易信息;

[0098] S18,放款申请流程结束。

[0099] 图5是本发明实施例的一种放款申请结果查询业务处理的流程图,如图5所示,其包括如下步骤:

[0100] S21,接收合作业务提交的放款结果查询交易;

[0101] S22,检查放款结果查询交易中是否存在放款申请号;若存在放款申请号,则执行S25;若不存在放款申请号,执行S23;

[0102] S23,查询是否存在明细交易数据,若存在明细交易数据,执行步骤S24;若不存在明细交易数据,记录交易时间和查询次数,执行S26;

[0103] S24,判断是否是超时交易,若不是超时交易,则此交易为查询失败交易,执行步骤S27,若是超时交易,返回超时异常,执行步骤S26;

[0104] S25,返回放款申请交易状态;

[0105] S26,返回正在放款中;

[0106] 本实施例中,正在放款中是指将交易要返回的json报文中交易结果标志设置为“放款中”。

[0107] S27,放款申请结果查询流程结束。

[0108] 图6是本发明实施例的一种放款申请结果查询业务处理的流程图,如图6所示,其包括如下步骤:

[0109] S31,放款申请回调:通用交易发起申请回调请求;

[0110] S32,动态交易处理:微服务的放款回调生产者动态处理回调交易;

[0111] 本实施例中,动态配置是指按照交易结果和交易类型对该业务交易进行协议转换,动态配置实质上是一个交易转换的过程。

[0112] S33,放款结果查询:进行放款结果信息处理;

[0113] 本实施例中,通过微服务接口调用,使用客户信息和放款申请编号在数据库中查询对应放款申请明细信息,取到对应的结果状态,将返回值按照接口要求传输给下一个处理过程。例如:放款结果查询,查询报文信息为(客户号:cust001,申请号:apply001,字段n...),通过客户编号查询到客户身份证号,再拿身份证号和申请号,在申请明细表中取放款申请的处理结果字段,最后将结果按照接口标准转换为成功、失败、异常等。

[0114] S34,放款结果交易:进行放款回调交易处理。

[0115] 本实施例中,根据S33查得的结果信息,组装成报文,通过微服务调用放款结果交易,由该交易处理后续业务逻辑(如:放款合同处理、借据生成、还款计划生成等)。

[0116] 具体的,本实施例在业务应用层设计开发通用交易API,保证存量交易和后续交易与此交易接口交互标准和方式在平台内标准统一,方式一致,申请交易处理完成后,由通用交易发起申请回调请求,通用交易本实施例的一个统一的接口,回调交易发起请求时使用放款申请交易在银行系统处理完成后,需要放款业务处理服务,作为消费者发起交易回调请求,将放款结果报文发送给放款申请的微服务。

[0117] 在一些实施例中,步骤S3中的根据业务处理结果和协议类型对回调接口进行交易动态配置以生成回调交易实例,具体包括:根据业务处理结果进行协议转换,根据转换后的协议创建或使用已创建的交易回调消费者;交易回调消费者将交易结果的报文发送给业务微服务的交易回调生产者;交易回调生产者根据交易类型、目标交易名称、协议类型、交易报文和业务处理结果生成回调交易实例。

[0118] 本实施例中,根据业务交易的解析和分发配置,确定要调用目标交易的协议类型,使用spring类型加载器及注解管理等工具,创建http交易或者Dubbo交易,实现业务交易的协议转换。根据交易类型、目标交易名称、协议类型、交易报文和业务处理结果,使用spring提供的类加载器生成微服务交易,可以减少业务交易代码的开发和发布以及业务微服务部署。

[0119] 图7是本发明实施例的一种分布式跨协议回调交易生成的流程图,如图7所示,本实施例在业务应用层设计开发通用交易API,保证存量交易和后续交易与此交易接口交互标准和方式在平台内标准统一,方式一致;在本实施例中,现有系统中已有的交易不用改造,新增的交易设计标准也不受影响。

[0120] 本实施例在远程过程调用层包括交易类型处理,业务交易处理,接口配置处理,分组匹配,版本匹配,回调交易处理等,具体如下:

[0121] 交易类型处理:本实施例在交易类型处理中区分交易回调时需要使用Dubbo交易还是http交易;即从交易报文中获取配置信息,根据所述配置信息确定交易回调时的交易类型;

[0122] 业务交易处理:在业务交易处理中完成需要处理的通用处理逻辑,通用处理逻辑是业务需求统一要求处理的过程。

[0123] 接口配置处理:本实施例在接口配置中根据业务处理的交易结果和交易类型,对回调接口进行动态配置;即通过分发配置信息,按照业务处理要求创建或使用已创建的回调交易实例,将结果报文发送给业务微服务;

[0124] 分组匹配:本实施例在分组匹配中回调交易管理根据业务逻辑对回调进行分组匹配;即按照业务逻辑要求,将回调交易对应关系配置在分发配置管理器中。

[0125] 版本匹配:在版本匹配中回调交易管理根据业务逻辑对回调进行版本匹配;

[0126] 回调交易处理:判断是否为首次调用,若不是首次调用,则直接分发并调用回调交易实例,若是首次调用,则进行回调交易生成过程;其中,回调交易生成过程具体包括:

[0127] 使用监听管理获取业务微服务的相关配置信息,并根据相关配置信息通过注解生成回调交易实例,并将回调交易实例以动态注册的方式注册到业务微服务的注册中心,即一般回调交易实例是启动时扫描应用程序注册到在业务微服务的注册中心的,而本实施例中的动态注册是指可以使用时进行注册,同时也可以将配置在数据库中的回调交易实例注册到注册中心。本实施例在Java中,监听管理通常涉及到事件监听器的使用,这些监听器用于监听特定事件(如按钮点击、鼠标移动、键盘输入等)的发生,并在事件发生时执行相应的操作,本发明中监听的是放款申请交易被调用的过程。注解生成是通过使用Spring注解来生成或配置java类。在Spring框架中,注解(Annotations)是大量使用的,它们用于减少XML配置文件的复杂性,并提供了一种声明性的方式来配置应用程序。

[0128] 本实施例在回调交易生成过程中,使用监听管理取得微服务相关的配置信息,并根据业务处理逻辑匹配的接口、分组和版本信息等相关配置信息,通过注解生成需要的回调交易实例,之后通过交易实例管理对交易进行微服务动态注册到注册中心以供业务交易使用;本发明实施例既可以在使用时进行注册,同时也可以将配置在数据库中的交易注册到注册中心。

[0129] 本实施例在保证交易的微服务管理与平台内其它微服务一致,可以复用现有远程控制层。现有远程控制层由微服务开发框架提供,所有微服务使用并遵守统一开发规范;复用指的是本发明实施例不会影响已有和新增微服务,目的是已有微服务不用改造,减少开发成本和发布次数。

[0130] 在正常情况下,微服务是启动时扫描应用程序注册到注册中心的,但是本发明可以使用时进行注册,同时也可以将配置在数据库中的交易注册到注册中心。保证交易的微服务管理与平台内其它微服务一致,复用现有远程控制层。现有远程控制层由微服务开发框架提供,所有微服务使用并遵守统一开发规范;复用指的是本发明不会影响已有和新增微服务;目的是已有微服务不用改造,减少开发成本和发布次数。

[0131] 本发明实施例提供两个交易接口,一个http协议的,一个Dubbo协议的,可以接收

不同协议的请求,进行不同协议的转换,同时支持Dubbo交易和http交易,可以快速融合两种框架下的微服务交易,统一管理业务交易的初始化、业务交易分发执行逻辑,新增交易可以配置扩展,本实施例通过对回调交易请求实例进行动态管理,减少了资源占用。

[0132] 本发明实施例应用在多种分布式微服务架构系统时,降低了交易事务设计和微服务改造范围及难度,减少软件技术研发人员的重复投入,提高了交易回调处理设计开发工作的质量,提升系统的扩展性,能满足运营业务快速变化的发展需求,从而降低了企业在业务运营系统建设中的资源投入,加速企业数字化转型建设的速度。

[0133] 图8是本发明实施例的一种分布式跨协议回调交易的生成装置结构框图,如图8所示,该装置100包括:

[0134] 交易请求接收和分发模块110,用于接收合作业务请求端发送的交易请求,根据合作业务请求端的接口信息将交易请求分发至对应的业务微服务;

[0135] 微服务处理模块120,用于根据交易请求从交易报文中获取业务交易配置信息,根据业务交易配置信息确定回调交易的交易类型和协议类型,并根据交易类型对交易报文进行业务处理,获得业务处理结果;

[0136] 动态配置模块130,用于根据业务处理结果和协议类型对回调接口进行交易动态配置以生成回调交易实例,并将回调交易实例注册到业务微服务的注册中心;

[0137] 匹配模块140,用于根据回调交易实例对回调交易进行分组和版本匹配,获得匹配结果,并根据匹配结果进行回调交易处理,获得回调交易处理的结果;

[0138] 回调交易生成模块150,用于根据回调交易处理的结果调用回调交易实例进行业务逻辑处理,对业务逻辑处理的结果和协议类型进行组装,完成回调交易生成和回调交易处理。

[0139] 图9是本发明实施例的另一种分布式跨协议回调交易的生成装置结构框图,如图9所示,本实施例提供的装置100'还包括:分发策略配置模块105,用于将业务交易之间的调用映射关系配置在数据库中,从数据库中读取业务交易间的调用映射关系,根据调用映射关系将业务交易匹配到需要调用的回调交易。

[0140] 具体细节请参考图1至图7所示的方法实施例。

[0141] 下面参考图10,其示出了适于用来实现本公开实施例的电子设备。

[0142] 本公开实施例中的终端设备可以包括但不限于诸如移动电话、笔记本电脑、数字广播接收器、PDA(个人数字助理)、PAD(平板电脑)、PMP(便携式多媒体播放器)、车载终端(例如车载导航终端)等等的移动终端以及诸如数字TV、台式计算机等等的固定终端。图10示出的电子设备仅仅是一个示例,不应对本公开实施例的功能和使用范围带来任何限制。

[0143] 如图10所示,电子设备可以包括处理装置(例如中央处理器、图形处理器等)801,其可以根据存储在只读存储器(ROM)802中的程序或者从存储装置808加载到随机访问存储器(RAM)803中的程序而执行各种适当的动作和处理。在RAM803中,还存储有电子设备操作所需的各种程序和数据。处理装置801、ROM802以及RAM803通过总线804彼此相连。输入/输出(I/O)接口805也连接至总线804。

[0144] 通常,以下装置可以连接至I/O接口805:包括例如触摸屏、触摸板、键盘、鼠标、摄像头、麦克风、加速度计、陀螺仪等的输入装置806;包括例如液晶显示器(LCD)、扬声器、振动器等的输出装置807;包括例如磁带、硬盘等的存储装置808;以及通信装置809。通信装置

809可以允许电子设备与其他设备进行无线或有线通信以交换数据。虽然图8示出了具有各种装置的电子设备,但是应理解的是,并不要求实施或具备所有示出的装置。可以替代地实施或具备更多或更少的装置。

[0145] 特别地,根据本公开的实施例,上文参考流程图描述的过程可以被实现为计算机软件程序。例如,本公开的实施例包括一种计算机程序产品,其包括承载在非暂态计算机可读介质上的计算机程序,该计算机程序包含用于执行流程图所示的方法的程序代码。在这样的实施例中,该计算机程序可以通过通信装置809从网络上被下载和安装,或者从存储装置808被安装,或者从ROM802被安装。在该计算机程序被处理装置801执行时,执行本公开实施例的方法中限定的上述功能。

[0146] 需要说明的是,本公开上述的计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质或者是上述两者的任意组合。计算机可读存储介质例如可以是——但不限于——电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。计算机可读存储介质的更具体的例子可以包括但不限于:具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机访问存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑磁盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本公开中,计算机可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。而在本公开中,计算机可读信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了计算机可读的程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。计算机可读信号介质还可以是计算机可读存储介质以外的任何计算机可读介质,该计算机可读信号介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。计算机可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于:电线、光缆、RF(射频)等等,或者上述的任意合适的组合。

[0147] 在一些实施方式中,客户端、服务器可以利用诸如HTTP(HyperTextTransferProtocol,超文本传输协议)之类的任何当前已知或未来研发的网络协议进行通信,并且可以与任意形式或介质的数字数据通信(例如,通信网络)互连。通信网络的示例包括局域网(“LAN”),广域网(“WAN”),网际网(例如,互联网)以及端对端网络(例如,ad hoc(端对端)网络),以及任何当前已知或未来研发的网络。

[0148] 上述计算机可读介质可以是上述电子设备中所包含的;也可以是单独存在,而未装配入该电子设备中。

[0149] 上述计算机可读介质承载有一个或者多个程序,当上述一个或者多个程序被该电子设备执行时,使得该电子设备:

[0150] 接收合作业务请求端发送的交易请求,根据合作业务请求端的接口信息将交易请求分发至对应的业务微服务;

[0151] 根据交易请求从交易报文中获取业务交易配置信息,根据业务交易配置信息确定回调交易的交易类型和协议类型,并根据交易类型对交易报文进行业务处理,获得业务处理结果;

[0152] 根据业务处理结果和协议类型对回调接口进行交易动态配置以生成回调交易实

例,并将回调交易实例注册到业务微服务的注册中心;

[0153] 根据回调交易实例对回调交易进行分组和版本匹配,获得匹配结果,并根据匹配结果进行回调交易处理,获得回调交易处理的结果;

[0154] 根据回调交易处理的结果调用回调交易实例进行业务逻辑处理,对业务逻辑处理的结果和协议类型进行组装,完成回调交易生成和回调交易处理。

[0155] 可以以一种或多种程序设计语言或其组合来编写用于执行本公开的操作的计算机程序代码,上述程序设计语言包括但不限于面向对象的程序设计语言—诸如Java、Smalltalk(一种面向对象的程序设计语言)、C++,还包括常规的过程式程序设计语言—诸如“C”语言或类似的程序设计语言。程序代码可以完全地在用户计算机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中,远程计算机可以通过任意种类的网络——包括局域网(LAN)或广域网(WAN)一连接到用户计算机,或者,可以连接到外部计算机(例如利用因特网服务提供商来通过因特网连接)。

[0156] 附图中的流程图和框图,图示了按照本公开各种实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分,该模块、程序段、或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个接连地表示的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意,框图和/或流程图中的每个方框、以及框图和/或流程图中的方框的组合,可以用执行规定的功能或操作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0157] 描述于本公开实施例中所涉及到的单元可以通过软件的方式实现,也可以通过硬件的方式来实现。其中,单元的名称在某种情况下并不构成对该单元本身的限定。

[0158] 本文中以上描述的功能可以至少部分地由一个或多个硬件逻辑部件来执行。例如,非限制性地,可以使用的示范类型的硬件逻辑部件包括:现场可编程门阵列(FPGA)、专用集成电路(ASIC)、专用标准产品(ASSP)、片上系统(SOC)、复杂可编程逻辑设备(CPLD)等等。

[0159] 在本公开的上下文中,机器可读介质可以是有形的介质,其可以包含或存储以供指令执行系统、装置或设备使用或与指令执行系统、装置或设备结合地使用的程序。机器可读介质可以是机器可读信号介质或机器可读储存介质。机器可读介质可以包括但不限于电子的、磁性的、光学的、电磁的、红外的、或半导体系统、装置或设备,或者上述内容的任何合适组合。机器可读存储介质的更具体示例会包括基于一个或多个线的电气连接、便携式计算机盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦除可编程只读存储器(EPROM或快闪存储器)、光纤、便捷式紧凑盘只读存储器(CD-ROM)、光学储存设备、磁储存设备、或上述内容的任何合适组合。

[0160] 以上描述仅为本公开的较佳实施例以及对所运用技术原理的说明。本领域技术人员应当理解,本公开中所涉及的公开范围,并不限于上述技术特征的特定组合而成的技术方案,同时也应涵盖在不脱离上述公开构思的情况下,由上述技术特征或其等同特征进行

任意组合而形成的其它技术方案。例如上述特征与本公开中公开的(但不限于)具有类似功能的技术特征进行互相替换而形成的技术方案。

[0161] 此外,虽然采用特定次序描绘了各操作,但是这不应理解为要求这些操作以所示出的特定次序或以顺序次序执行来执行。在一定环境下,多任务和并行处理可能是有利的。同样地,虽然在上面论述中包含了若干具体实现细节,但是这些不应被解释为对本公开的范围的限制。在单独的实施例的上下文中描述的某些特征还可以组合地实现在单个实施例中。相反地,在单个实施例的上下文中描述的各种特征也可以单独地或以任何合适的子组合的方式实现在多个实施例中。

[0162] 尽管已经采用特定于结构特征和/或方法逻辑动作的语言描述了本主题,但是应当理解所附权利要求书中所限定的主题未必局限于上面描述的特定特征或动作。相反,上面所描述的特定特征和动作仅仅是实现权利要求书的示例形式。

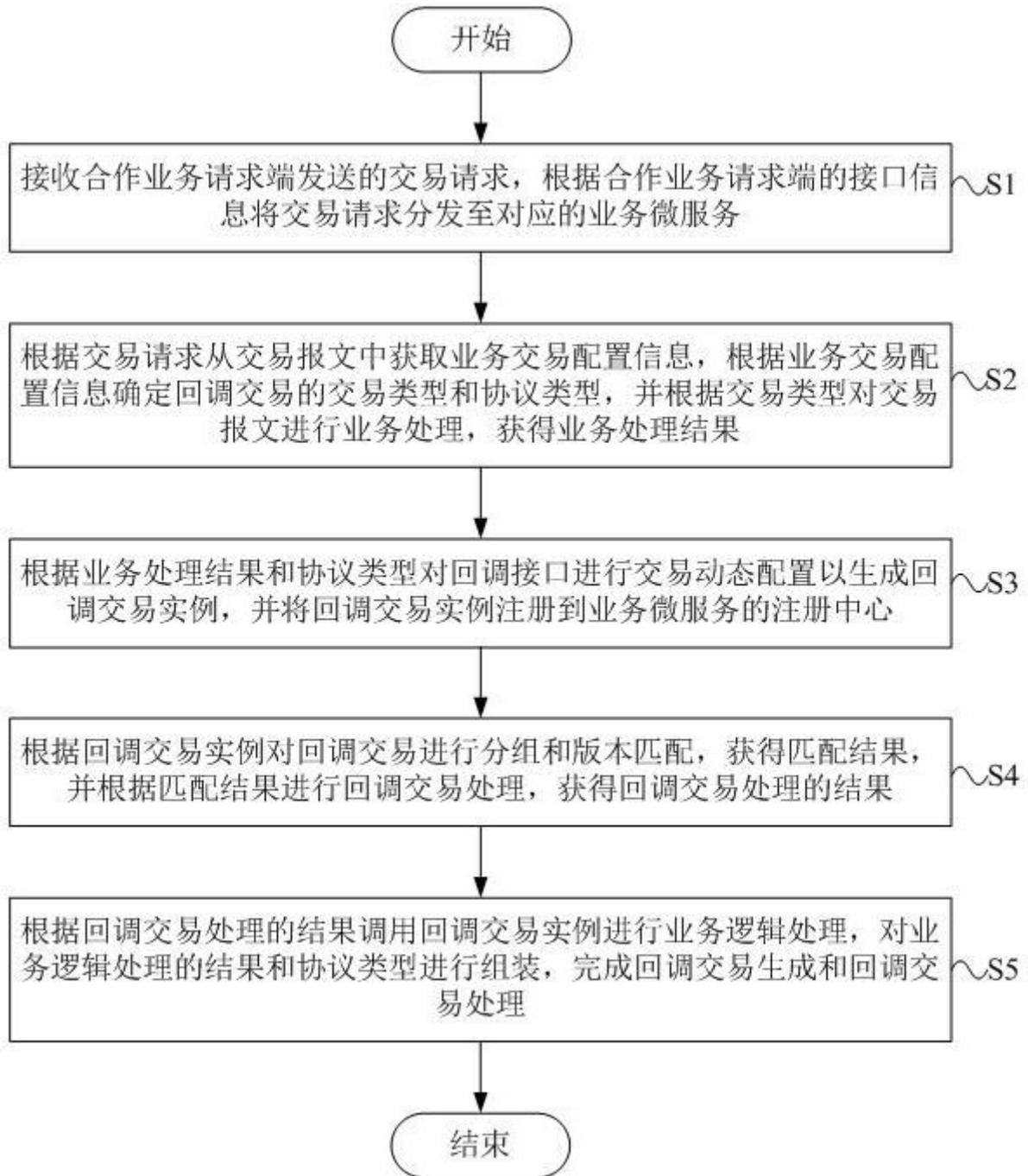


图 1

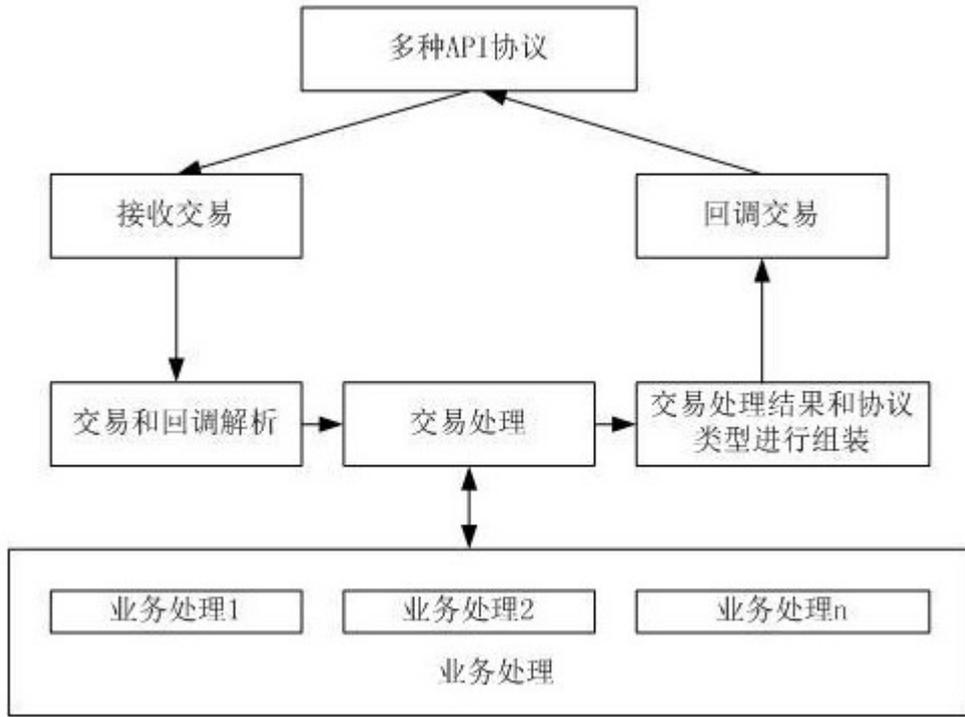


图 2

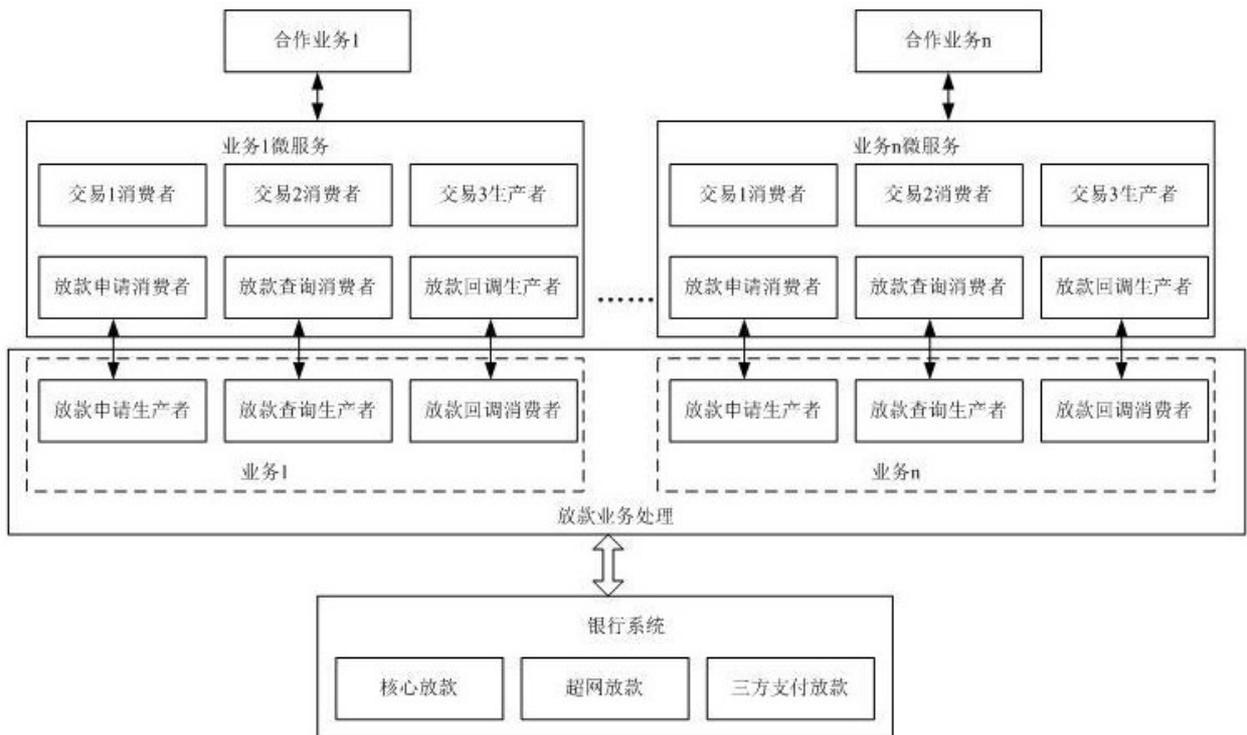


图 3

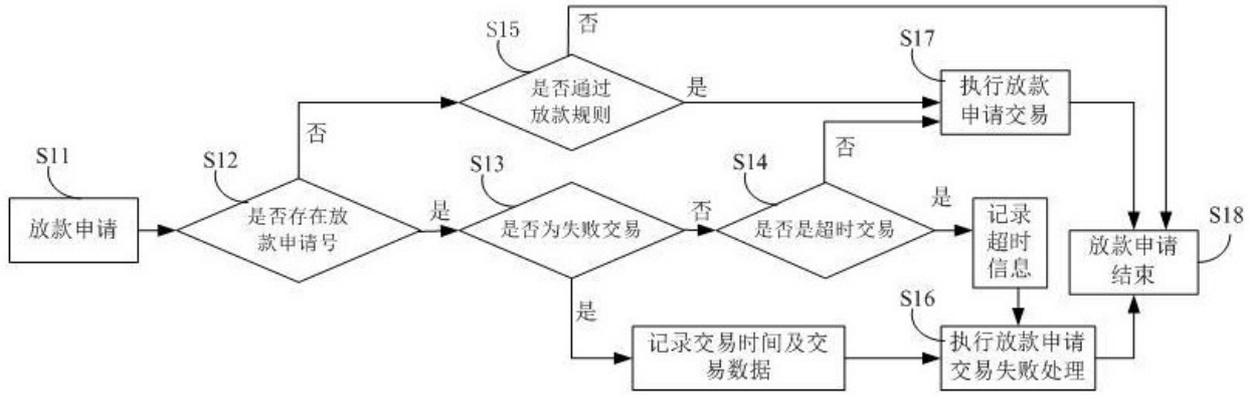


图 4

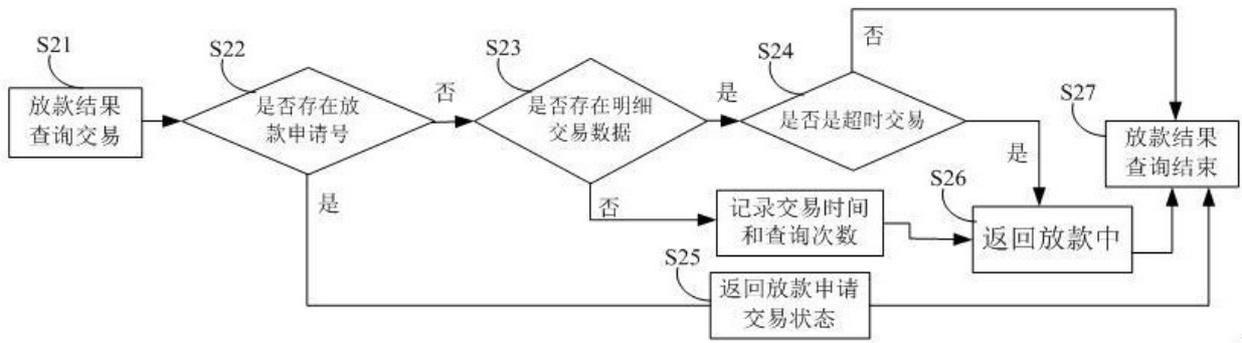


图 5



图 6

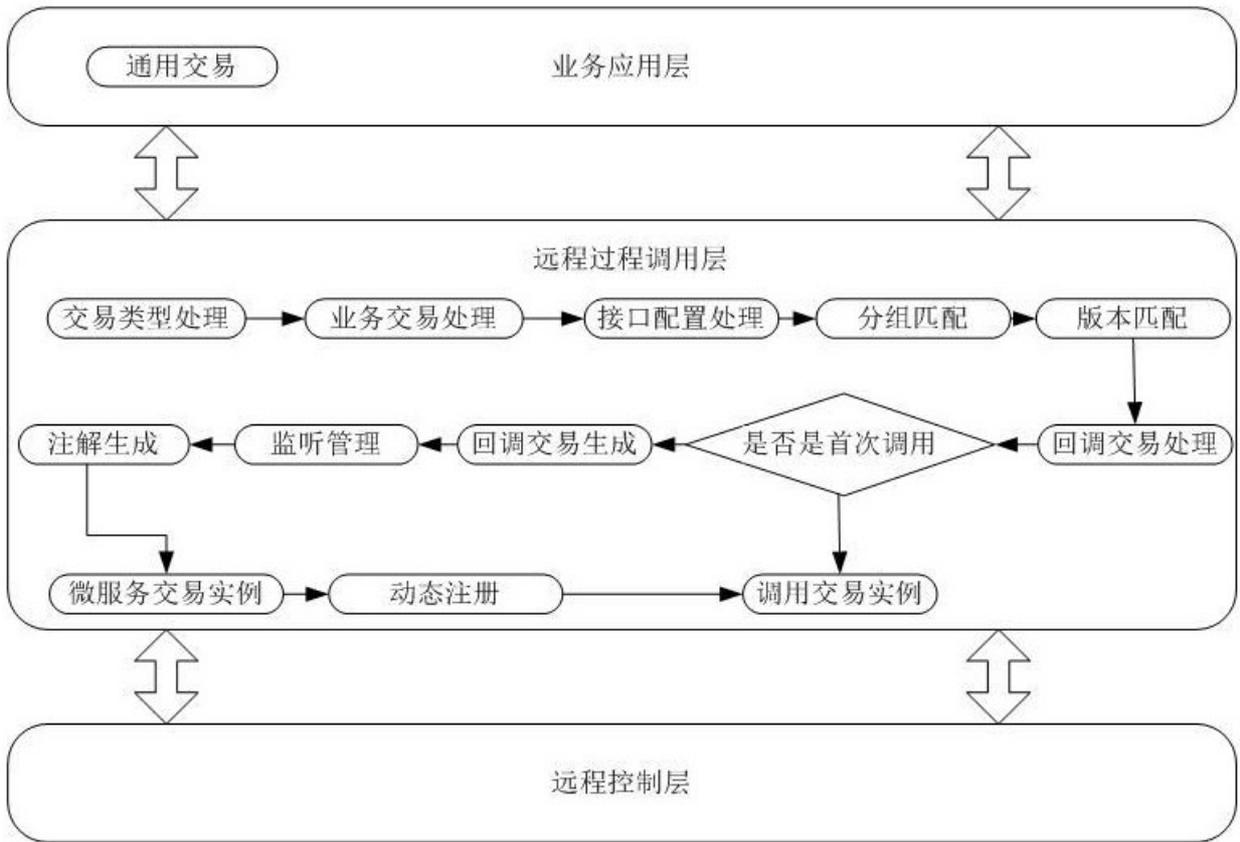


图 7

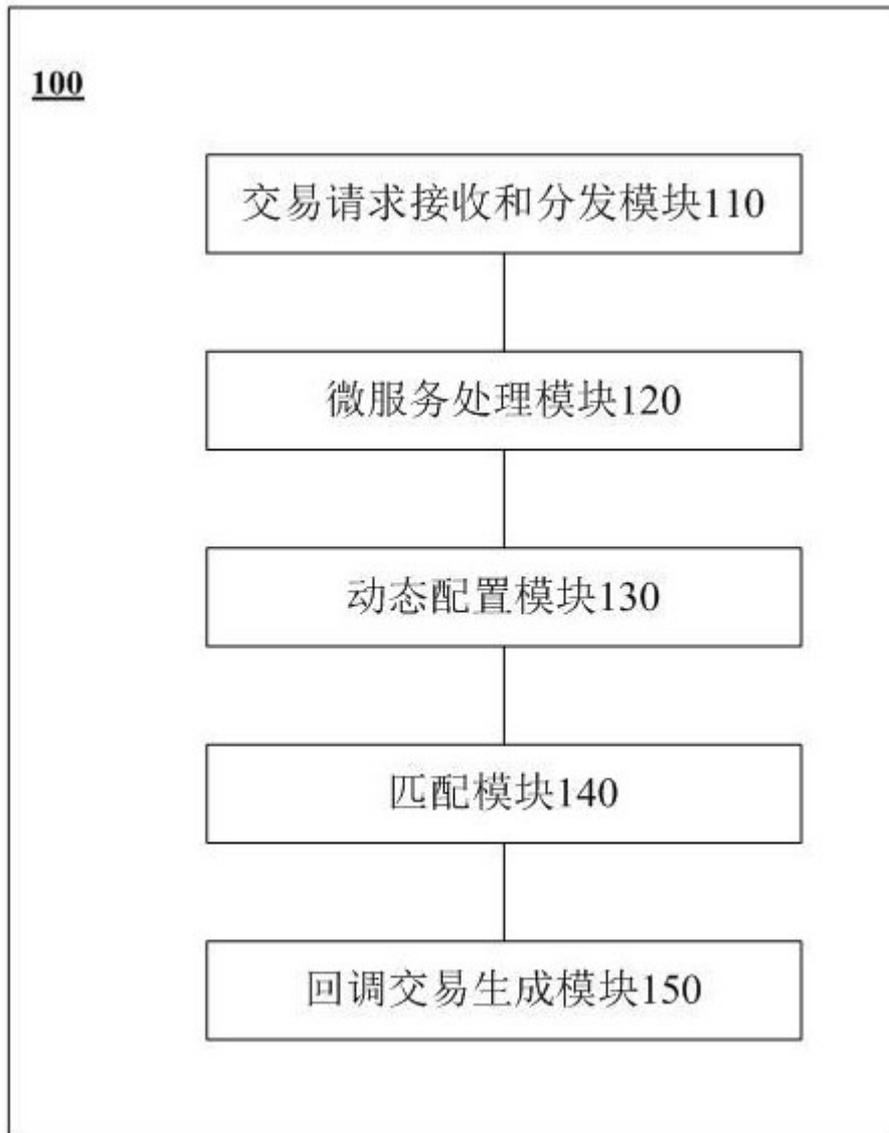


图 8

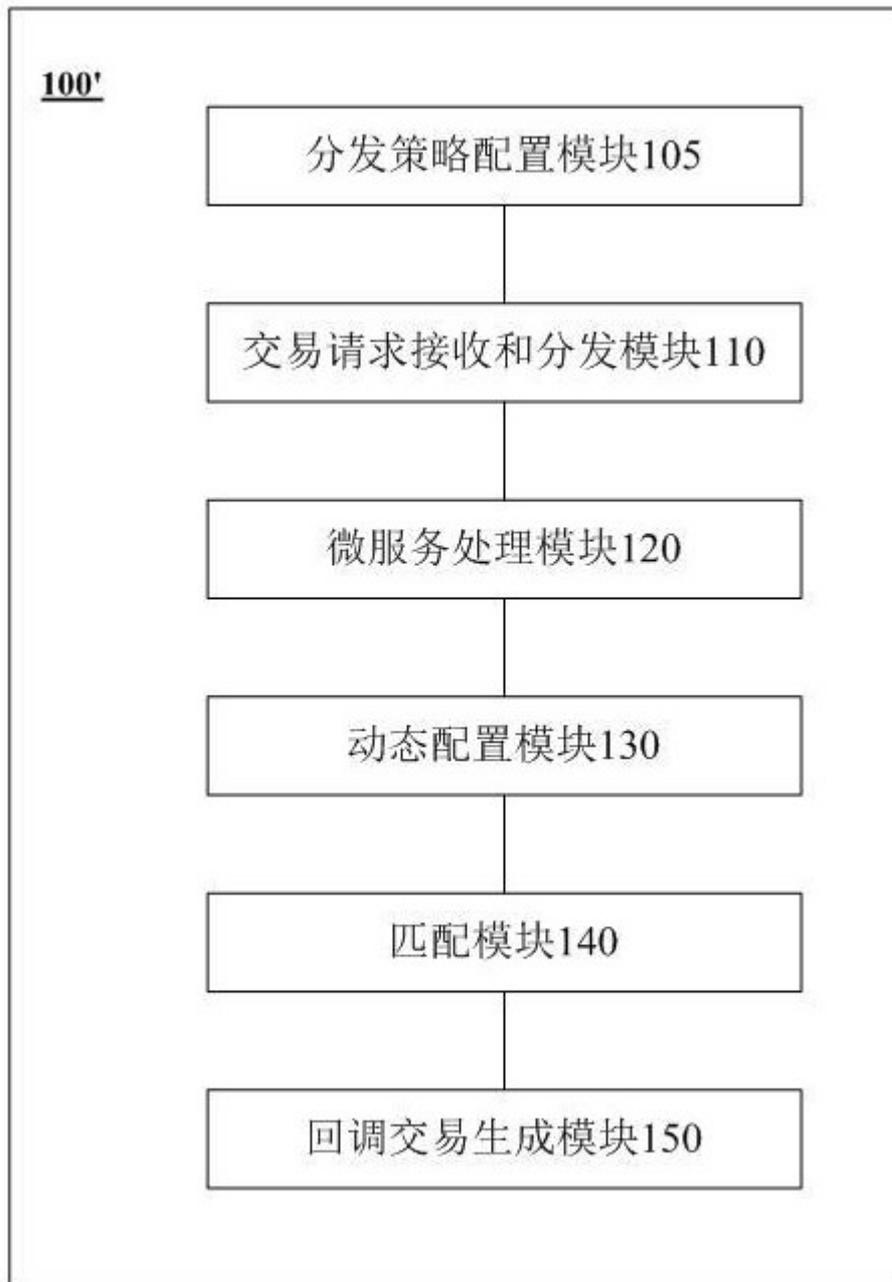


图 9

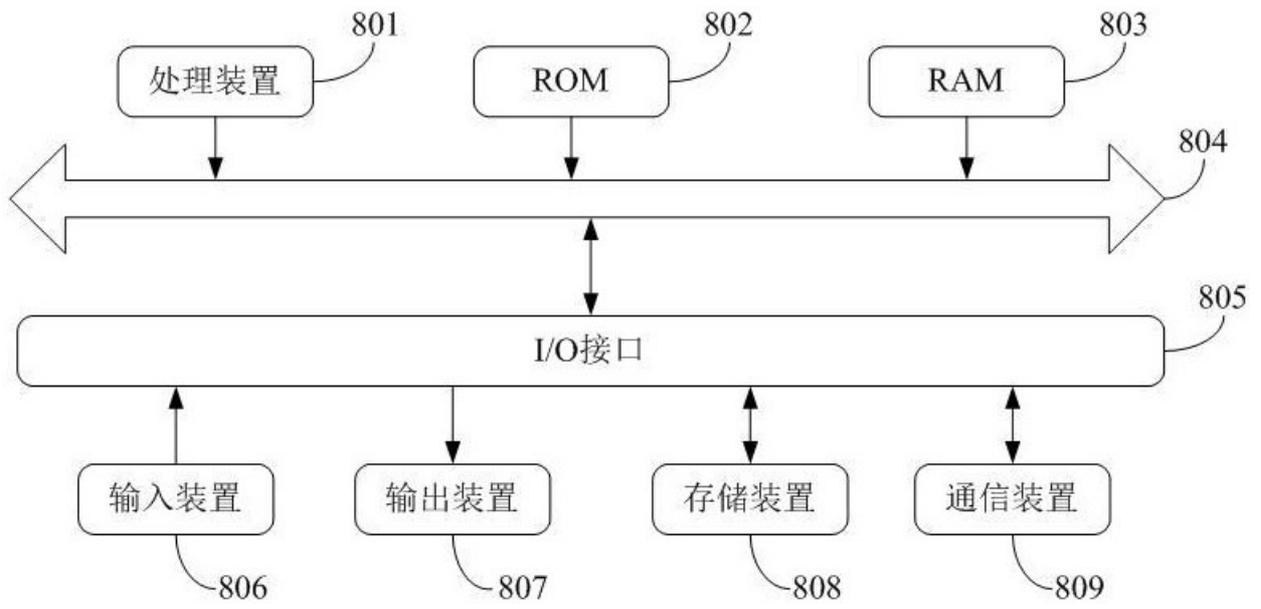


图 10