(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 106445824 A
(43)申请公布日 2017.02.22

(21)申请号 201610875057.5

(22)申请日 2016.09.30

(71)申请人 南京途牛科技有限公司
地址 210042 江苏省南京市玄武区玄武大道699-32号途牛大厦

(72)发明人 吴海祥

(74)专利代理机构 南京众联专利代理有限公司 32206
代理人 叶涓涓

(51)Int.Cl.
G06F 11/36(2006.01)
G06F 9/44(2006.01)

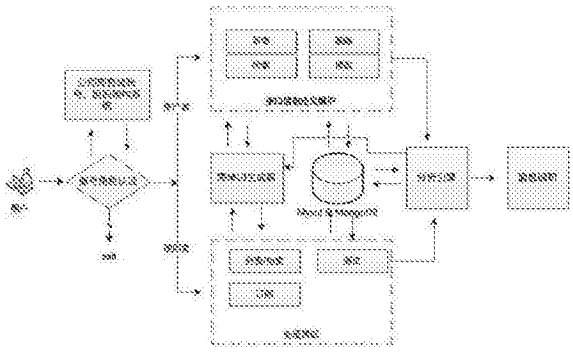
权利要求书2页 说明书8页 附图5页

(54)发明名称

一种基于角色的接口综合性管理系统

(57)摘要

本发明公开了一种基于角色的接口综合性管理系统，包括：用户角色管理模块、接口信息的录入与维护模块、表单UI生成器、验证测试模块、分析引擎、消息通知模块，所述用户角色管理模块用于管理用户信息并对用户登录进行认证；所述接口信息录入与维护模块用于对每个接口的基本属性、入参和出参等信息进行入库和管理；所述使用者角色的验证测试模块用于根据当前入参请求指定环境接口，经过分析引擎分析后导出详细的分析测试结果，如有异常则利用消息通知模块自动通知关联人；消息通知模块用于实现接口异常通知功能和变更通知功能。



CN 106445824 A

1.一种基于角色的接口综合性管理系统，其特征在于，包括：用户角色管理模块、接口信息的录入与维护模块、表单UI生成器、验证测试模块、分析引擎、消息通知模块，所述用户角色管理模块用于管理用户信息并对用户登录进行认证；所述接口信息录入与维护模块用于对每个接口的基本属性、入参和出参等信息进行入库和管理；所述使用者角色的验证测试模块用于根据当前入参请求指定环境接口，经过分析引擎分析后导出详细的分析测试结果，如有异常则利用消息通知模块自动通知关联人；消息通知模块用于实现接口异常通知功能和变更通知功能；

所述接口信息录入与维护模块

用于输入和修改接口基本信息、入参、出参和测试桩，并进行保存，

用于调用表单UI生成器根据入参和出参生成所有节点定义表单，入参或出参中每个字段当做一个节点：利用表单UI生成器，支持用户手工添加节点，定义节点属性信息的方式维护接口定义信息；或通过入参或出参的json数据结构，调用分析引擎得到节点分析指标项，以及节点位置信息，并自动生成所有节点定义表单；

所述分析引擎包括分配器、各个子类型分析器和校验对比单元，所述分配器用于对参数进行解析，使json数据中的“对象”类型能原样输出为object类型而非array类型，在数据解析过程中保证类型信息不丢失，获得经过解析的对象的各个属性类型，根据类型的不同分发到各自的分析器中；所述各子类型分析器用于获取节点信息，获得每个节点在json数据结构中的唯一位置和该节点的所有分析指标项；所述校验对比单元用于将分析结果与之前保存的关于该接口的定义信息进行对比；

所述验证测试模块包括测试单元和订阅单元，测试单元用于将当前入参请求指定环境接口，经过分析引擎分析，导出详细的分析测试结果，如有异常则通过消息通知模块自动周知关联人，所述订阅单元将使用方感兴趣的接口记录与用户进行关联，订阅成功后，当接口定义发生变更或测试发现异常可及时通知关联用户。

2.根据权利要求1所述的基于角色的接口综合性管理系统，其特征在于：所述用户角色管理模块包括系统自管单元和第三方用户管理单元，所述系统自管单元由本系统自己实现用户的注册、登录、组织结构等信息的录入与维护，所述第三方用户管理单元用于调用第三方用户管理模块进行用户权限认证、用户和组织结构基本信息的获取。

3.根据权利要求1所述的基于角色的接口综合性管理系统，其特征在于：所述节点类型分为三类：根节点、标量节点和非标量节点，非标量节点支持添加子节点、删除本节点、节点属性维护、子节点展开和收起。

4.根据权利要求3所述的基于角色的接口综合性管理系统，其特征在于：所述每个节点的html结构均为动态浮动型，支持无限扩展。

5.根据权利要求1所述的基于角色的接口综合性管理系统，其特征在于：所述分配器解析时利用php语言中的json_decode (string $json [, bool $assoc = false])方法对参数进行解析，保留该方法的第二位参数$assoc为false，使json数据中的对象类型能原样输出为object类型而非array类型。

6.根据权利要求5所述的基于角色的接口综合性管理系统，其特征在于：所述分配器将经过解析的object对象通过get_object_vars方法将对象属性转换成关联数组，再继续通过遍历，使用gettype方法获取各个属性类型，根据类型的不同分发到各自的分析器中。

7.根据权利要求1或5所述的基于角色的接口综合性管理系统，其特征在于：所述子类型分析器包括：boolean类型分析器、double类型分析、integer类型分析器、string类型分析器、object类型分析器、array类型分析器、null类型分析器。

8.根据权利要求1所述的基于角色的接口综合性管理系统，其特征在于：所述验证测试模块还包括所述列表检索单元，列表检索单元根据接口关键字或所属部门信息检索出目标接口。

一种基于角色的接口综合性管理系统

技术领域

[0001] 本发明属于互联网技术领域，具体涉及一种基于角色的接口综合性管理系统。

背景技术

[0002] 伴随着互联网和移动互联网技术的快速发展，各种应用程序层出不穷。而不管是应用程序本身还是应用程序背后的支撑系统内部，各个子系统、模块之间的通信都是通过大量的接口调用实现，因此应用程序的开发及使用过程中，不可避免地会调用大量接口。现有研发流程中对接口的管理主要采用以下如下方式。

[0003] 1.功能测试：通过手工附带参数请求接口，人工一层层查看接口出参是否符合约定来验证接口功能；

2.文档定义：接口文档定义通过开发人员手工编辑文档，一个字段一个字段地维护约定信息；

3.消息告知：当接口定义发生变更或接口测试发现问题时通过人工口头告知或邮件方式向各使用方解释变动细节；

由于目前的研发流程中缺乏一套对接口进行综合性管理的系统或方法。在大量接口应用过程中，普遍存在如下问题：

1.接口文档的定义，每次都需要由生产方手动维护编辑，缺乏一套标准的接口定义术语并常常伴随定义粒度不足和准确性有误等问题；

2.开发联调前，缺少对全量接口的自动化测试，一个庞大系统上线，待测试接口成百上千，都由测试人员手工一个个接口测试，测试覆盖率和准确性都存在巨大隐患，严重影响开发效率和项目整体进度的推进；

3.接口定义变更时不能及时更新文档并通知接口使用方，造成联调时进度受阻；

4.接口测试中发现问题时不能第一时间由系统自动通知接口生产方；

5.接口维护方发生人员变动时无法及时平滑地完成交接。

发明内容

[0004] 为解决上述问题，本发明公开了一种基于角色的接口综合性管理系统，实现接口的角色归属，统一维护接口定义，对接口进行全粒度地自动化测试，自动化生成标准化的接口定义文档和测试报告，当接口定义变更或测试发现问题时能够由系统自动通知关联人。

[0005] 为了达到上述目的，本发明提供如下技术方案：

基于角色的接口综合性管理系统，包括：用户角色管理模块、接口信息的录入与维护模块、表单UI生成器、验证测试模块、分析引擎、消息通知模块，所述用户角色管理模块用于管理用户信息并对用户登录进行认证；所述接口信息录入与维护模块用于对每个接口的基本属性、入参和出参等信息进行入库和管理；所述使用者角色的验证测试模块用于根据当前入参请求指定环境接口，经过分析引擎分析后导出详细的分析测试结果，如有异常则利用消息通知模块自动通知关联人；消息通知模块用于实现接口异常通知功能和变更通知功

能；

所述接口信息录入与维护模块

用于输入和修改接口基本信息、入参、出参和测试桩，并进行保存，

用于调用表单UI生成器根据入参和出参生成所有节点定义表单，入参或出参中每个字段当做一个节点：利用表单UI生成器，支持用户手工添加节点，定义节点属性信息的方式维护接口定义信息；或通过入参或出参的json数据结构，调用分析引擎得到节点分析指标项，以及节点位置信息，并自动生成所有节点定义表单；

所述分析引擎包括分配器、各个子类型分析器和校验对比单元，所述分配器用于对参数进行解析，使json数据中的“对象”类型能原样输出为object类型而非array类型，在数据解析过程中保证类型信息不丢失，获得经过解析的对象的各个属性类型，根据类型的不同分发到各自的分析器中；所述各子类型分析器用于获取节点信息，获得每个节点在json数据结构中的唯一位置和该节点的所有分析指标项；所述校验对比单元用于将分析结果与之前保存的关于该接口的定义信息进行对比。

[0006] 所述验证测试模块包括测试单元和订阅单元，测试单元用于将当前入参数请求指定环境接口，经过分析引擎分析，导出详细的分析测试结果，如有异常则通过消息通知模块自动周知关联人，所述订阅单元将使用方感兴趣的接口记录与用户进行关联，订阅成功后，当接口定义发生变更或测试发现异常可及时通知关联用户。

[0007] 进一步的，所述用户角色管理模块包括系统自管单元和第三方用户管理单元，所述系统自管单元由本系统自己实现用户的注册、登录、组织结构等信息的录入与维护，所述第三方用户管理单元用于调用第三方用户管理模块进行用户权限认证、用户和组织结构基本信息的获取。

[0008] 进一步的，所述节点类型分为三类：根节点、标量节点和非标量节点，非标量节点支持添加子节点、删除本节点、节点属性维护、子节点展开和收起。

[0009] 进一步的，所述每个节点的html结构均为动态浮动型，支持无限扩展。

[0010] 进一步的，所述分配器解析时利用php语言中的json_decode (string $json [, bool $assoc = false])方法对参数进行解析，保留该方法的第二位参数$assoc为false，使json数据中的对象类型能原样输出为object类型而非array类型。

[0011] 进一步的，所述分配器将经过解析的object对象通过get_object_vars方法将对象属性转换成关联数组，再继续通过遍历，使用gettype方法获取各个属性类型，根据类型的不同分发到各自的分析器中。

[0012] 进一步的，所述子类型分析器包括：boolean类型分析器、double类型分析、integer类型分析器、string类型分析器、object类型分析器、array类型分析器、null类型分析器。

[0013] 进一步的，所述验证测试模块还包括所述列表检索单元，列表检索单元根据接口关键字或所属部门信息检索出目标接口。

[0014] 与现有技术相比，本发明具有如下优点和有益效果：

通过对接口出、入参数中层层字段类型和值限定范围做最精细粒度的自动化解析，实现了基于用户角色对接口实行分层管理和问题跟踪；通过根据出、入参自动化生成图形化的接口定义录入表单，方便用户对复杂接口结构做出定义维护。同时，本发明利用MongoDb

数据库对接口分析结果进行快速存取，并通过消息模块实现当检测到接口定义发生变更或验证测试接口发现问题时实时通知接口关联人。本发明提供的管理系统能够显著降低接口开发维护成本，提高整体研发效率。

附图说明

[0015] 图1为系统数据流程图。

[0016] 图2为字段节点html结构设计图。

[0017] 图3为分析引擎逻辑流程图。

[0018] 图4为Boolean类型分析器结构图。

[0019] 图5为double类型分析器结构图。

[0020] 图6为integer类型分析器结构图。

[0021] 图7为object类型分析器结构图。

[0022] 图8为array类型分析器结构图。

[0023] 图9为分析引擎中分配器、各类型分析器的具体处理流程图。

[0024] 图10为消息通知模块流程图。

具体实施方式

[0025] 以下将结合具体实施例对本发明提供的技术方案进行详细说明，应理解下述具体实施方式仅用于说明本发明而不用于限制本发明的范围。

[0026] 本发明提供的基于角色的接口综合性管理系统包括如下几部分：用户角色管理模块、接口信息的录入与维护模块、表单UI生成器、验证测试模块、分析引擎、消息通知模块。其中各模块之间数据流程图如图1所示。用户角色管理模块用于管理用户信息(新增、修改、删除)并对用户登录进行认证；接口信息录入与维护模块用于对每个接口的基本属性、入参和出参等信息进行入库和管理；使用者角色的验证测试模块用于检索目标接口，根据当前入参数请求指定环境接口，经过分析引擎分析后导出详细的分析测试结果，如有异常则利用消息通知模块自动通知关联人；消息通知模块用于配合使用者角色的验证测试模块实现接口异常通知功能和变更通知功能。

[0027] 用户角色管理模块主要包括两个部分：一是系统内自管的一套用户角色、组织结构维护单元(以下简称“系统自管单元”)；二是第三方用户管理单元，该单元用于调用第三方用户管理模块，如各个公司内的OA系统等。

[0028] 系统自管单元由本系统自己实现用户的注册、登录、组织结构等信息的录入与维护。其中组织结构数据由系统超级管理员录入，组织结构层级支持无限分级；由用户在注册界面填写基本信息，如用户名、密码、邮箱、手机号码、所属部门等信息，提交保存入库，完成注册；用户在登录界面填写账号和密码信息提交服务端验证，验证通过即可进入本系统，登录用户可在个人中心维护编辑用户信息，超级管理员可管理所有用户基本信息和组织结构数据。

[0029] 第三方用户管理单元无需在本系统内维护用户和组织结构信息，而是利用第三方系统公开的接口完成用户权限认证、用户和组织结构基本信息的获取。具体实现时，用户在登录界面填写账号信息，提交系统后，由系统将此信息发送第三方账号验证系统(如公司域

账号系统)，如验证成功则在本系统session中标示用户登录成功。除验证用户权限外，第三方用户管理单元还应提供三方面功能，一是支持通过用户名联想相关用户，二是根据用户ID 和部门ID 获取相应的详细信息，三是能提供组织结构全量信息数据；这些功能均通过与本系统调用的第三方用户管理模块进行交互实现。

[0030] 登录系统后可选择接口生产者和使用者这两种角色中的其中一种，每一种角色对应不同的功能区域，生产者对应接口信息与录入模块，使用者则对应验证测试模块。选择不同角色即可使用相应模块。

[0031] 接口信息录入与维护模块主要包括以下几个子功能单元：新增、删除、检索、修改(编辑)，这些功能均在列表页操作。列表页上有新增按钮，列表页支持按照维护者或部门检索，找出由本人或本部门提供支持的接口列表，每条记录右侧有编辑和删除两个操作按钮，点击“编辑”按钮可对该接口的基本信息进行维护更新，“删除”按钮则直接删除该接口信息；其中新增子单元用于新增接口数据，具体包括接口基本信息、入参、出参和测试桩。接口信息录入与维护模块能够根据接口数据，利用表单UI生成器生成所有节点定义表单。在列表点击“新增”按钮，则进入接口信息维护表单页，该表单页中包括基本信息TAB页、入参和出参TAB页和测试桩TAB页。基本信息TAB页主要由接口名、维护者(默认当前登录用户)、请求方式、路由地址、所属系统三字码、各环境对应的地址等字段组成。在入参和出参TAB页上利用表单UI生成器，用户可通过手工一个个添加节点，定义节点属性信息的方式维护接口定义信息，也可通过现成的入参或出参的json数据结构，通过“一键生成”的功能调用表单UI生成器自动生成所有节点定义表单；在测试桩TAB页上可维护该接口多套测试桩数据，方便不同订阅者使用。在列表点击“编辑”按钮，同样进入接口信息维护表单页，页面上TAB页同新增功能，各字段读取当前需要修改的接口数据并加以显示。在新增或编辑之后，系统接受前台表单传递的数据，将接口基础信息和自动生成的入参、出参表单整体结构数据保存msyql数据表api_info表中，将经过表单UI生成器得到的入参、出参定义的信息以接口名为唯一键值保存进MongoDB中。

[0032] 使用者角色的验证测试模块包括列表检索单元、测试单元和订阅单元。通过列表检索单元，使用者可使用接口关键字或所属部门信息检索出目标接口。测试单元提供测试功能页面，在该页面上可使用该生产者提供的多套测试桩数据，也支持使用者输入自定义测试数据。点击页面中的“测试”按钮，系统会将当前入参请求指定环境接口，经过分析引擎分析(获得入参、出参中各字段(节点)信息，将分析结果与之前接口定义进行比对分析)，导出详细的分析测试结果，如有异常则通过消息通知模块自动周知关联人。而订阅单元可将使用方感兴趣的接口记录与用户进行关联，并及时通知使用者订阅记录的变更信息。通过点击接口记录右侧的订阅按钮，即可在“我的订阅”单元中查看已关注的全部接口记录，订阅接口后，有关该接口的任何变更信息都会以邮件或短信的方式告知使用者，确保使用方第一时间获取接口变更信息。

[0033] 表单UI生成器将用户已有的json数据传入后端，由服务端脚本php分析数据结构，一层层遍历解析节点属性信息生成表单的html结构，并将默认节点的解析数据保存进入js对象中输出前端，生成的表单结构和节点属性信息，用户可在前端根据需要继续微调，对表单进行进一步扩展。其主要包括两方面功能，一是根据接口预设计文档，通过人工的方式支持创建结构化的表单，将书面的接口定义维护进系统；二是支持导入现有的接口入参、出参

智能生成结构化录入表单。

[0034] 具体的说，本发明将入参或出参中每个字段都当做一个节点，并将节点类型分为三类：根节点、标量节点和非标量节点。非标量节点支持添加子节点、删除本节点、节点属性维护、子节点展开和收起。表单UI生成器获得入参、出参后，调用分析引擎进行分析，获得入参、出参中各字段(节点)分析指标项，以及各字段(节点)位置信息，根据这些节点信息生成表单。

[0035] 上述每个节点的html结构均设计为动态浮动型，支持无限扩展，节点结构如图2所示，节点定义的信息以悬浮表单控件的方式维护，根据不同的节点类型，维护有所不同。每个节点的基础信息均有字段名称、值类型，而不同节点类型关于值的限定有所不同，例如boolean类型值限定真、假或无限制，integer类型限定值大小范围：大于零、小于零或等于零，可多选；double类型可限定小数点位数等等，这些具体的值限定可根据实际项目需求继续添加扩展。

[0036] 前述的分析引擎通过遍历将父节点位置一层层往下传递，直到目标节点，最终获得位置信息及为该节点在JSON数据结构中的唯一位置，通过该位置可用于唯一确定并获取该节点所有信息，并能够获得节点类型指标。分析引擎包括分配器、各个子类型分析器和校验对比单元。其中分配器用于对参数进行解析，使json数据中的对象类型能原样输出为object类型而非array类型，在数据解析过程中保证类型信息不丢失，获得经过解析的对象的各个属性类型，根据类型的不同分发到各自的分析器中。各子类型分析器用于获取节点信息，获得每个节点在json数据结构中的唯一位置和该节点的所有分析指标项。校验对比单元用于将分析结果与之前保存的关于该接口的定义信息进行对比。具体流程如图3所示。

[0037] 分配器首先接收外部json类型入参，利用解析JSON数据结构的方法对参数进行解析。本发明实施例中利用php语言中的json_decode (string $json [, bool $assoc = false])方法对参数进行解析，保留该方法的第二位参数$assoc为false，使json数据中的对象类型能原样输出为object类型而非array类型，确保在数据解析过程中类型信息不丢失。

[0038] 随后，分配器将经过解析的object对象通过get_object_vars方法将对象属性转换成关联数组，再继续通过遍历，使用gettype方法获取各个属性类型，根据类型的不同分发到各自的分析器中，获得分析结果集合主要应用在表单UI生成器中根据分析结果自动生成接口定义的录入表单；另外在使用者角色验证测试接口时，将分析结果与之前接口定义进行比对分析，逐条列出参数中各个节点项的分析结果，如遇与定义不一致则列出异常并通过消息系统通知接口关联人。

[0039] 本发明根据目标json数据类型创建各个子类型分析器：boolean类型分析器、double类型分析、integer类型分析器、string类型分析器、object类型分析器、array类型分析器、null类型分析器。各子类型分析器用于获取节点信息并与节点定义信息进行比较判断，经过分析器的处理，能够获得每个节点在json数据结构中的唯一位置和该节点的所有分析指标项。Boolean类型分析器的分析指标项有限定真、限定假、不限，分析器结构如图4所示；double类型支持限定小数点位数，指标项有是否限定小数点位数、小数点位数值、正则表达式，将实际返回结果与定义中的这些指标进行比较，不匹配则给出明细信息；分析器结构如图5所示；integer类型支持大于零、小于零、等于零、不限、正则表达式几项指标，将

实际返回结果与定义中的这些指标进行比较，不匹配则给出明细信息；分析器结构如图6所示；string类型支持分析自定义正则规则、是否为空；object类型分析器根据子项的类型继续分发到其他类型分析器中，分析器结构如图7所示，分析指标为是否为空；array类型分析器分析指标有判断为空、item个数，并将每个子项目分发到object类型分析器中继续分析，分析器结构如图8所示。各类型的分析指标可自由添加，并不受本例具体指标限制。

[0040] 校验对比单元利用以上步骤中保存的接口定义信息对象，先将该对象转换成数组类型，对数组元素进行遍历，因为每个元素的定义结果都是以它本身的唯一位置为键值保存的，所以可根据此位置在实际的接口参数中快速定位寻找：

1.根据字段在定义中是否“必须含有”，“必须含有”的字段而在实际返回中未找到，则直接抛出异常，存在则继续往下判断；

2.判断目标字段与定义中的类型信息是否匹配，如不匹配也直接抛出异常，类型匹配则继续往下判断；

以下举例说明分析引擎中分配器、各类型分析器的具体处理流程，流程图如图9所示。

[0041] 假设入参、出参的json格式的数据结构如下：

```
{
"productId":210122643,
"canSell":true,
"price":123.69,
    "adultQuantity" :2,
    "childQuantity" : 1,
   "segmentList":[
        {
          "dCityCode" : "2500",
          "aCityCode" : "785095" ,
          "departDate" : "2016-08-30"
        }
    ],
    "flightSessionId":"3c1018c7094329800162a6ad2f4833a3",
}
```

该参数中含有integer、double、array、object和string类型，分配器获取当前节点类型，根据不同类型分流到各自类型分析器中进行处理；

我们以Object类型分析器为例，其内部处理流程具体如下：

a)获取当前节点键值和类型(根节点键值为空，可设置一个初始化值)作为object类型的原始分析结果，以节点位置为键值压入“节点分析堆栈”中如图⑤；

b)以当前节点为id，并计算当前节点在表单展现偏左的距离位置，根节点初始化偏左距离可为0，子元素偏左距离每一层往下递增，具体递增幅度可自由设定，生成节点表单html代码，也以节点位置为键值压入“表单元素堆栈”中如图④；

c)Object类型作为复合类型，在进行往下遍历之前，先将对象属性转换成数组元素；

d)遍历中获取每个子元素的偏左距离，每往里一层便递增一次，同一层中的节点偏左

距离相等;结合父元素和子元素的键值获取每个子节点在整个数据结构中的唯一位置;将这些信息连同子元素本身带入分配器②中重新分流处理;

e)待子元素全部分析处理完毕,需要对object类型的表单元素进行收尾,压入一个冗余的尾部表单节点元素到“表单元素堆栈”中,该对象属性信息和b)步奏中基本类似,只是展现内容不一样,在object类型节点中的所有子元素在表单呈现上能呈现一个归属整体的效果;

integer类型分析器、string类型分析器、boolean类型分析器、double类型分析、object类型分析器、null类型分析器对于这些标量节点的分析具体处理步骤与object类型分析中的a、b步骤相同:

a)获取当前节点键值和类型(根节点键值为空,可设置一个初始化值)作为object类型的原始分析结果,以节点位置为为键值压入节点分析堆栈;

b)以当前节点为id,并计算当前节点在表单展现偏左的距离位置,根节点初始化偏左距离可为0,生成节点表单html代码,也以节点位置为键值压入表单元素堆栈中;

array类型分析器的处理流程同object类型的a、b、d、e步骤,只是中间省略步奏c,因其本身即为array类型无需转换。

[0042] 在一次接口分析过程中,利用所有节点压入的表单元素堆栈能够生成UI表单,供用户操作编辑更新节点属性信息,将用户自定义信息维护到节点分析堆栈中,将最终的各个节点的定义信息入库。

[0043] 分析引擎主要应用在两个地方,一是在表单UI生成器中,接收录入的入参、出参分析各字段属性信息用于初始化表单结构;二是在验证测试接口时,对实际请求接口的入参、出参各字段结构和值进行分析,并与接口定义信息进行比较,将结果格式化导出,并将异常信息通过消息系统周知接口关联人。

[0044] 消息通知模块功能流程如图10,其主要用于实现两种功能——异常通知功能和变更通知功能。异常通知功能是当接口订阅者检验分析接口时发现问题或异常实时通知生产者和订阅者;当用户以使用者角色进入系统,需要分析检测某个接口时,由分析引擎对测试桩入参和接口实际返回的出参进行分析,将分析结果与之前维护的接口定义进行比较,如发现异常,则通知消息模块,消息模块通过查询对应接口的生产者信息,通过邮件或短信的方式及时告知生产者异常明细,当生产者处理完毕时又重新反馈周知到订阅者。这里当检测发现异常时同时将对应接口和异常插入待处理列表中,由系统脚本定期检测异常清单,发现超过一定期限未处理的接口,则由消息模块通过查询组织结构信息,获取生产者所在部门经理或上级信息,将异常信息告知经理及时处理。

[0045] 本发明还进一步利用系统定时器Crontab工具实现定期检查接口问题处理进度,超过一定时限未跟进处理的接口,则由系统直接通知该接口生产者的直属上级,跟进尽快解决。

[0046] 变更通知功能是生产者对接口定义信息进行变更时,系统可自动将变更信息邮件通知各使用方。当接口生产者由于业务需要对接口定义信息进行变更时,系统会自动分析出变更明细信息,将变更信息和对应的唯一接口名传入消息系统,消息模块通过接口名查询该接口订阅者信息(主要是联系方式),将变更明细信息通过邮件的方式周知所有订阅者,并自动生成最新的接口定义文档方面用户查阅。

[0047] 本发明方案所公开的技术手段不仅限于上述实施方式所公开的技术手段，还包括由以上技术特征任意组合所组成的技术方案。应当指出，对于本技术领域的普通技术人员来说，在不脱离本发明原理的前提下，还可以做出若干改进和润饰，这些改进和润饰也视为本发明的保护范围。

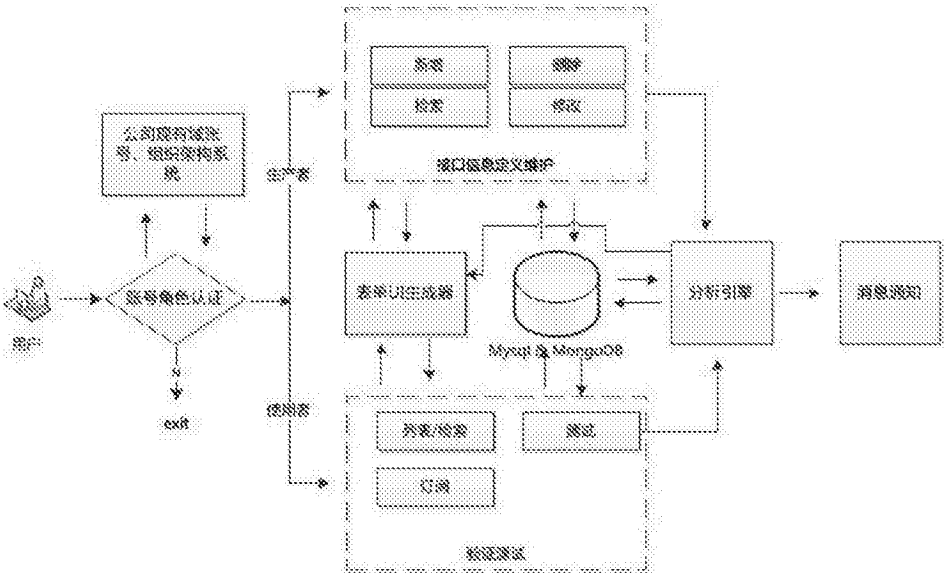


图1

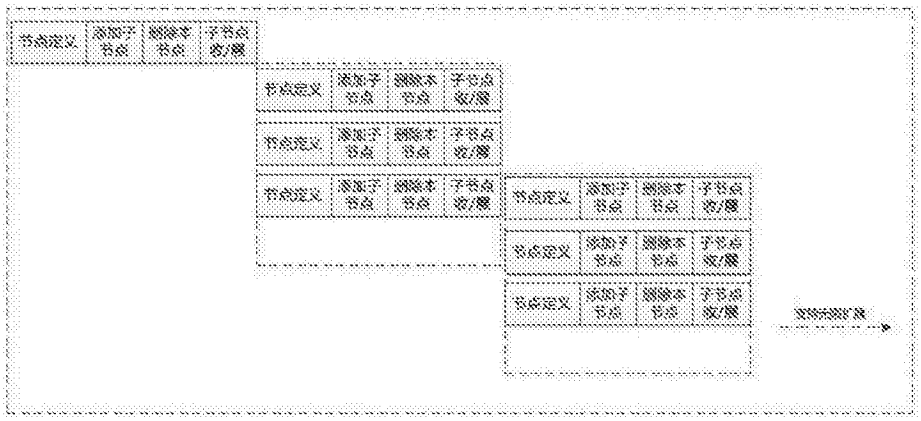


图2

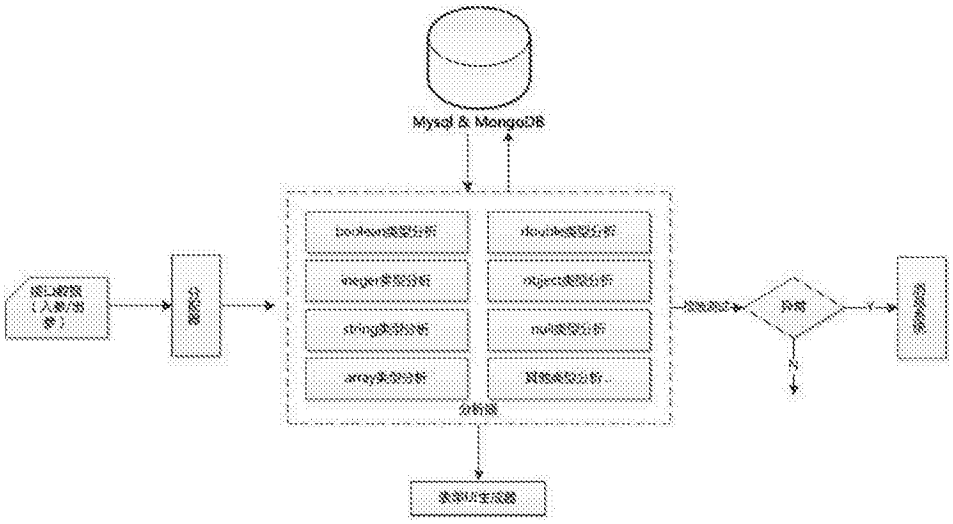

Mysql & MongoDB

图3

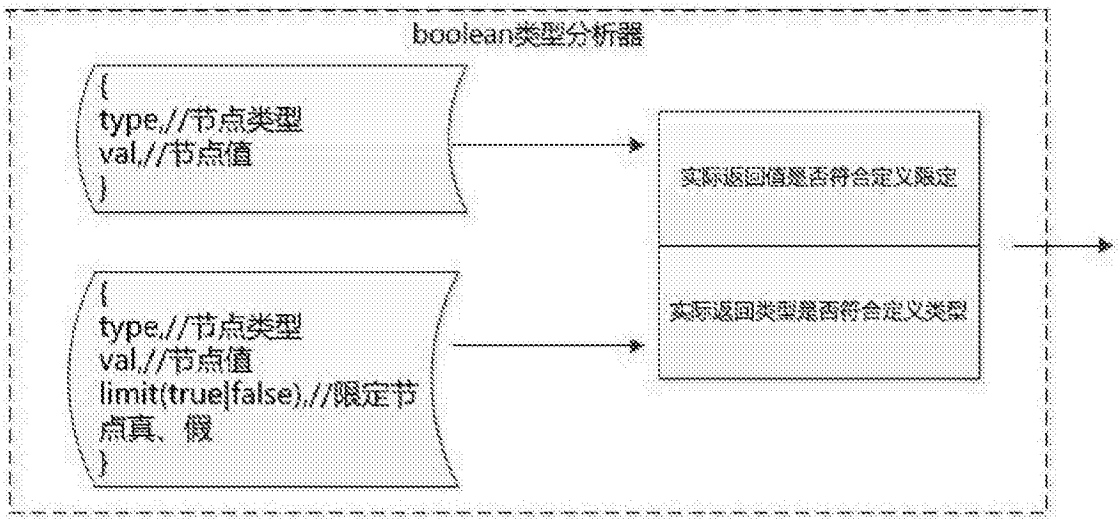

boolean类型分析器
{
type,//节点类型
val,//节点值
}
{
type,//节点类型
val,//节点值
limit(true|false),//限定节点真、假
}
实际返回值是否符合定义限定
实际返回类型是否符合定义类型

图4

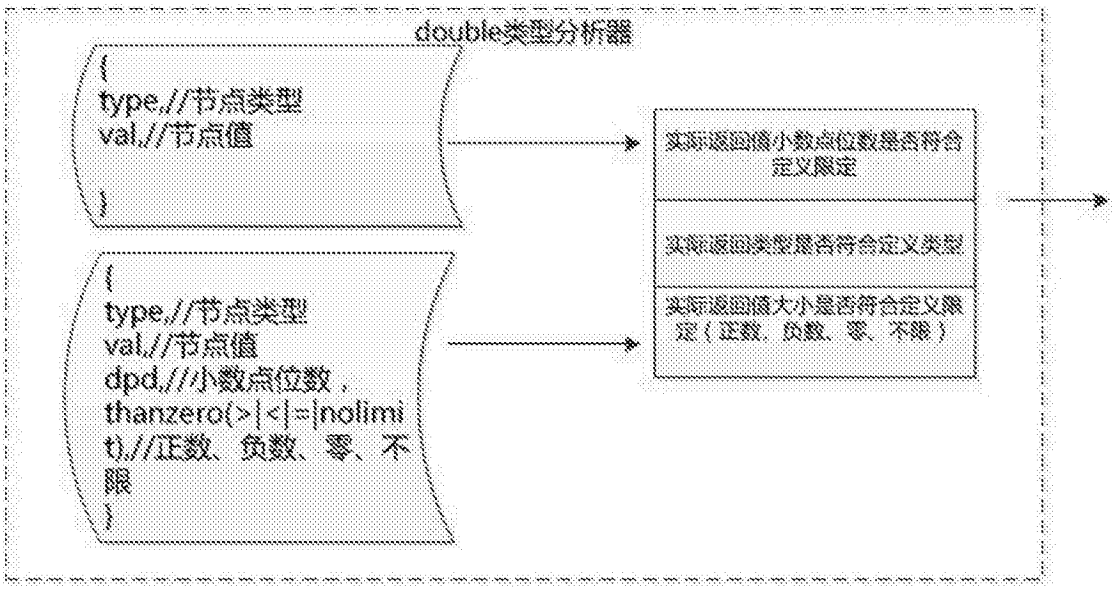

double类型分析器
{
type,//节点类型
val,//节点值
}
{
type,//节点类型
val,//节点值
dpd,//小数点位数，
thanzero(>|<|=|nolimit),//正数、负数、零、不限
}
实际返回值小数点位数是否符合定义限定
实际返回类型是否符合定义类型
实际返回值大小是否符合定义限定（正数、负数、零、不限）

图5

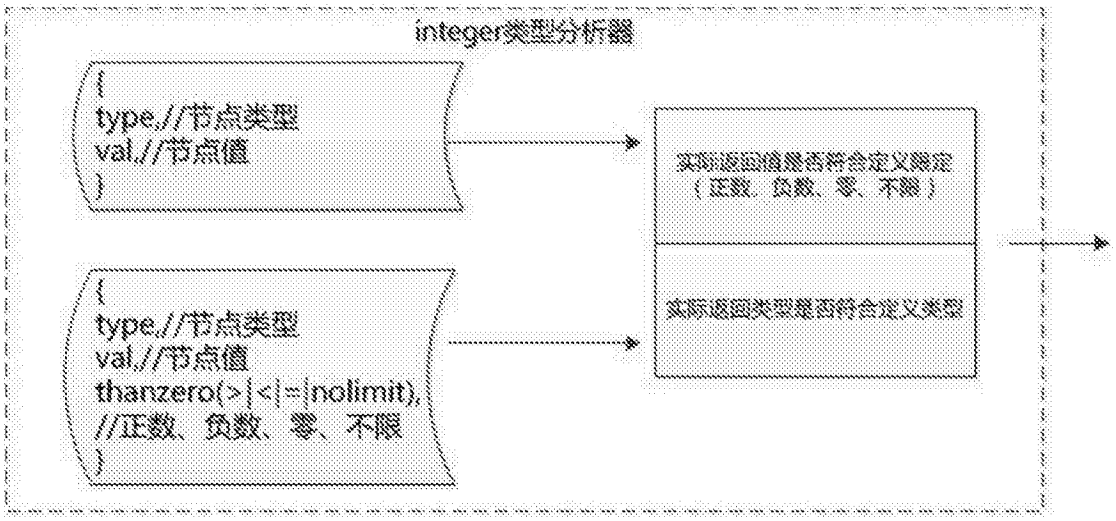

integer类型分析器
{
type,//节点类型
val,//节点值
}
{
type,//节点类型
val,//节点值
thanzero(>|<|=|nolimit),
//正数、负数、零、不限
}
实际返回值是否符合定义限定（正数、负数、零、不限）
实际返回类型是否符合定义类型

图6

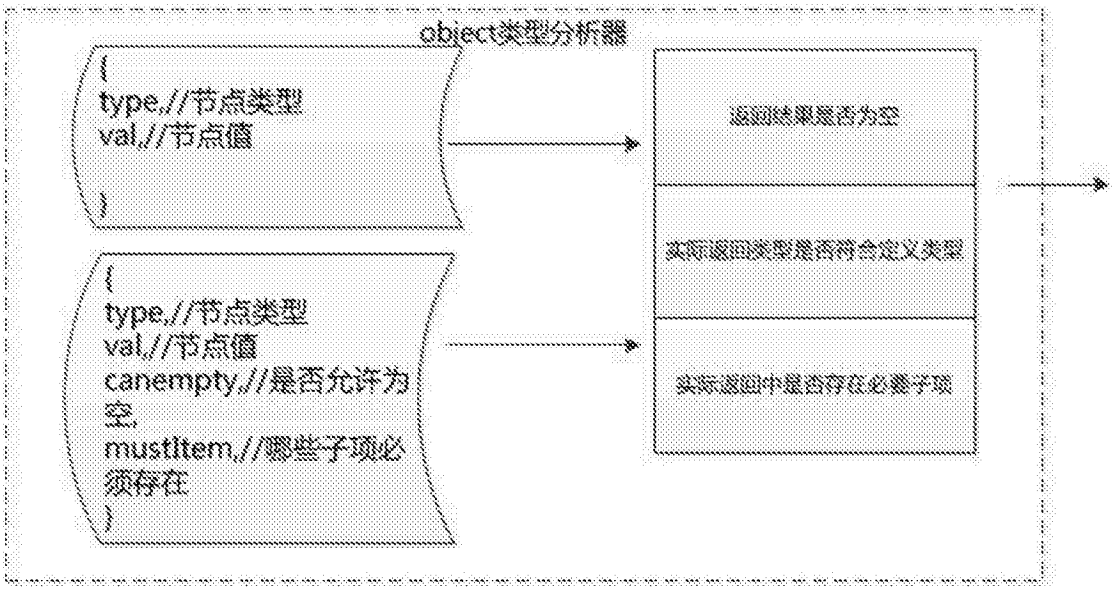

object类型分析器
{
type,//节点类型
val,//节点值
}
{
type,//节点类型
val,//节点值
canempty,//是否允许为空,
mustItem,//哪些子项必须存在
}
返回结果是否为空
实际返回类型是否符合定义类型
实际返回中是否存在必要子项

图7

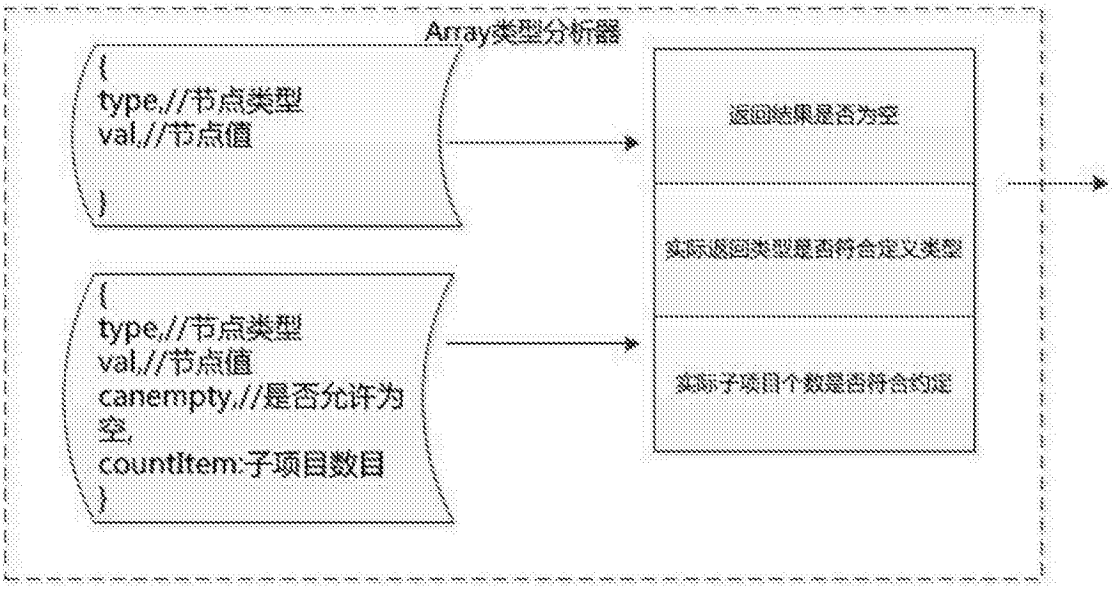

Array类型分析器
{
type,//节点类型
val,//节点值
}
{
type,//节点类型
val,//节点值
canempty,//是否允许为空,
countItem:子项目数目
}
返回结果是否为空
实际返回类型是否符合定义类型
实际子项目个数是否符合约定

图8

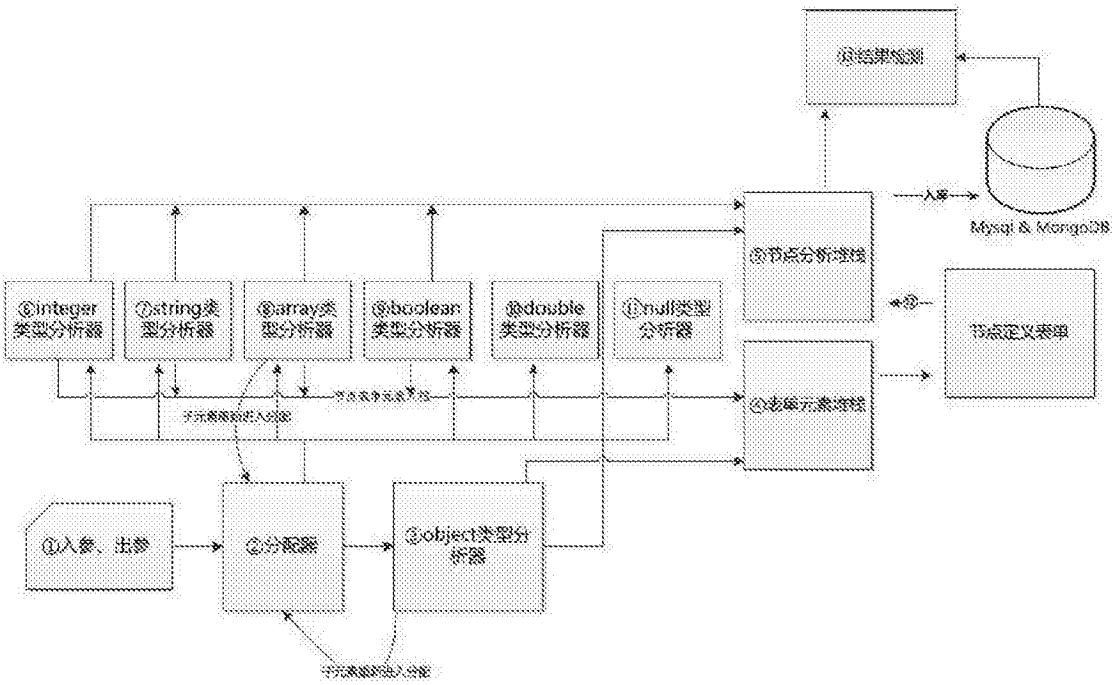

⑥integer
类型分析器
⑦string类
型分析器
⑧array类
型分析器
⑨boolean
类型分析器
⑩double
类型分析器
⑪null类型
分析器
③节点分析主线
②主线元素遍线
节点定义表单
Mysql & MongoDB
①入参、出参
②分配器
③object类型分
析器

图9

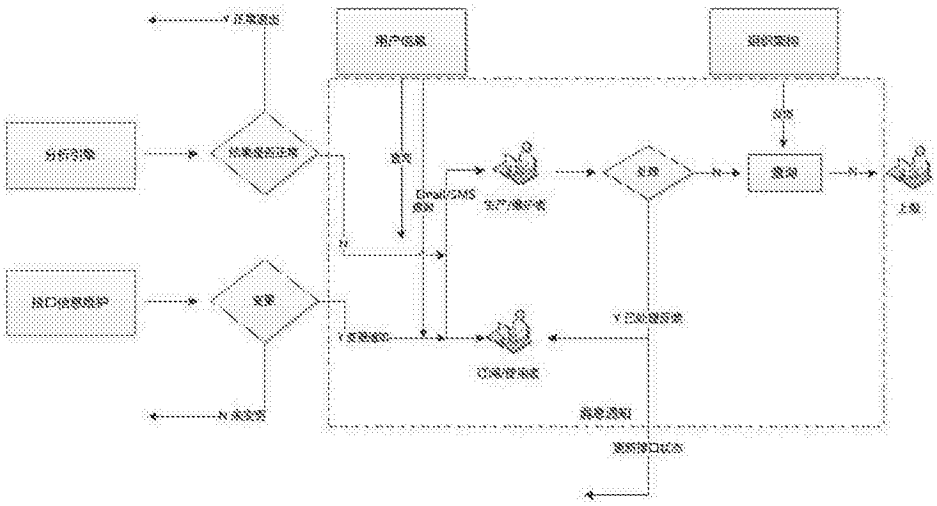


图10