



(12) 发明专利

(10) 授权公告号 CN 102096631 B

(45) 授权公告日 2013. 09. 25

(21) 申请号 200910254327. 0

(22) 申请日 2009. 12. 11

(73) 专利权人 华为技术有限公司

地址 518129 广东省深圳市龙岗区坂田华为
总部办公楼

(72) 发明人 陈志利 杨达宇

(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 彭愿洁 李文红

(51) Int. Cl.

G06F 11/36(2006. 01)

(56) 对比文件

CN 101488085 A, 2009. 07. 22, 全文.

CN 101145993 A, 2009. 03. 19, 全文.

CN 101110024 A, 2009. 01. 23, 全文.

DE 10137297 A1, 2004. 02. 20, 全文.

CN 101025686 A, 2007. 08. 29, 全文.

审查员 张帅领

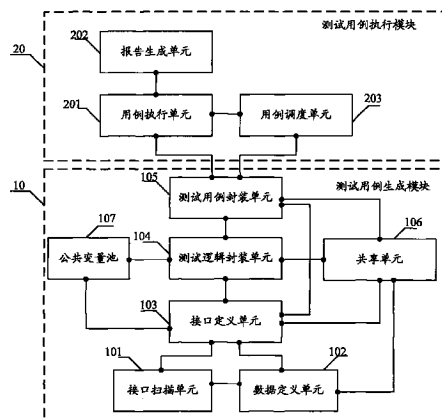
权利要求书5页 说明书17页 附图15页

(54) 发明名称

接口测试方法及装置

(57) 摘要

本发明提供一种接口测试方法及装置,其接口测试装置包括:接口扫描单元,用于根据用户指定的接口参数类型扫描被测试对象;数据定义单元,用于分析接口扫描单元对被测试对象的扫描结果,展现接口参数类型的结构,接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息;测试用例封装单元,用于获取用户选择的用例元素,利用数据定义单元获取的用例元素中基本对象类型的配置信息,获取各基本对象类型对应的测试数据,利用用例元素及其中各基本对象类型对应的测试数据封装测试用例;用例执行单元,用于运行测试用例;报告生成单元,用于根据测试用例的运行结果,生成测试报告。使用本发明实施例提供的技术方案,能够提高测试效率。



1. 一种接口测试装置,包括测试用例封装单元、用例执行单元和报告生成单元,所述用例执行单元,用于运行所述测试用例封装单元封装的测试用例;所述报告生成单元,用于根据测试用例的运行结果,生成测试报告,其特征在于,所述接口测试装置还包括:接口扫描单元、数据定义单元,

所述接口扫描单元,用于根据用户指定的接口参数类型扫描被测试对象;

所述数据定义单元,用于分析所述接口扫描单元对所述被测试对象的扫描结果,展现所述接口参数类型的结构,所述接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息;

所述测试用例封装单元,用于获取用户选择的用例元素,利用所述数据定义单元获取的用例元素中基本对象类型的配置信息,获取所述用例元素中各基本对象类型对应的测试数据,利用所述用例元素及其中各基本对象类型对应的测试数据封装测试用例。

2. 根据权利要求1所述的测试装置,其特征在于,

所述数据定义单元,用于分析所述接口扫描单元对所述被测试对象的扫描结果,当所述接口参数类型为复杂对象类型时,展现所述接口参数类型的树形结构,获取用户指定的所述接口参数类型中各基本对象类型的配置信息,根据各基本对象类型的配置信息和接口参数类型的树形结构配置接口参数对象;当所述接口参数类型为基本对象类型时,获取用户指定的所述基本对象类型的配置信息,根据所述基本对象类型的配置信息配置接口参数对象。

3. 根据权利要求2所述的测试装置,其特征在于,

所述接口测试装置还包括:

接口定义单元,用于获取用户指定的接口寻址信息、接口返回类型和接口参数类型,封装接口,所述接口包括:接口寻址信息,接口返回类型对应的接口返回对象,所述数据定义单元配置的接口参数对象;

所述测试用例封装单元获取的用例元素包括:接口定义单元封装的接口。

4. 根据权利要求3所述的测试装置,其特征在于,还包括:共享单元,

所述数据定义单元,还用于在接收到用户指示将接口参数类型的配置信息共享的指令后,将接口参数类型的配置信息保存到共享单元;所述接口参数类型的配置信息包括接口参数类型中各基本对象类型的配置信息;

所述共享单元,用于保存所述接口参数类型的配置信息;其中,所述共享单元中保存的接口参数类型中各基本对象类型的配置信息是所述数据定义单元以后配置接口参数对象时用户所指定的基本对象类型的配置信息;

或者,

所述接口定义单元,还用于接收用户指示将接口共享的指令,将所述接口保存到共享单元;

所述共享单元,用于保存所述接口;其中,所述共享单元保存的接口是测试用例封装单元以后封装测试用例时用户所选择的用例元素;

或者,

所述测试用例封装单元,还用于接收用户指示将测试用例共享的指令,将所述测试用例保存到共享单元;

所述共享单元,用于保存所述测试用例。

5. 根据权利要求 2 所述的测试装置,其特征在于,

所述接口测试装置还包括:

接口定义单元,用于获取用户指定的接口寻址信息、接口返回类型和接口参数类型,封装接口,所述接口包括:接口寻址信息,接口返回类型对应的接口返回对象,所述数据定义单元配置的接口参数对象;

测试逻辑封装单元,用于获取用户选择的多个逻辑封装元素,按照用户指定的调用关系将多个逻辑封装元素封装成测试逻辑结构,所述多个逻辑封装元素包括所述接口定义单元封装的接口;

所述测试用例封装单元获取的用例元素包括:测试逻辑封装单元封装的测试逻辑结构。

6. 根据权利要求 5 所述的测试装置,其特征在于,

所述多个逻辑封装元素包括第一逻辑封装元素和第二逻辑封装元素;

所述测试逻辑封装单元,用于获取用户选择的多个逻辑封装元素,将第一逻辑封装元素的返回结果作为变量,根据用户的指令设置变量的值作为第二逻辑封装元素的输入值,得到测试逻辑结构。

7. 根据权利要求 5 所述的测试装置,其特征在于,还包括:共享单元,

所述数据定义单元,还用于在接收到用户指示将接口参数类型的配置信息共享的指令后,将接口参数类型的配置信息保存到共享单元;所述接口参数类型的配置信息包括接口参数类型中各基本对象类型的配置信息;

所述共享单元,用于保存所述接口参数类型的配置信息;其中,所述共享单元保存的接口参数类型中各基本对象类型的配置信息是所述数据定义单元以后配置接口参数对象时用户所指定的基本对象类型的配置信息;

或者,

所述接口定义单元,还用于接收用户指示将接口共享的指令,将所述接口保存到共享单元;

所述共享单元,用于保存所述接口;其中,所述共享单元保存的接口是测试用例封装单元以后封装测试用例时用户所选择的用例元素或者是测试逻辑封装单元以后封装测试逻辑结构时用户所选择的逻辑封装元素;

或者,

所述测试逻辑封装单元,还用于接收用户指示将测试逻辑结构共享的指令,将所述测试逻辑结构保存到共享单元;

所述共享单元,用于保存所述测试逻辑结构;其中,所述共享单元保存的测试逻辑结构是测试用例封装单元以后封装测试用例时用户所选择的用例元素;

或者,

所述测试用例封装单元,还用于接收用户指示将测试用例共享的指令,将所述测试用例保存到共享单元;

所述共享单元,用于保存所述测试用例。

8. 根据权利要求 4 或者 7 所述的测试装置,其特征在于,

所述共享单元,还用于将所保存的内容发送给中央控制器。

9. 根据权利要求 1 所述的测试装置,其特征在于,

所述测试用例封装单元包括:

测试数据获取子单元,用于展现用例元素中具有配置信息的基本对象类型,根据用户的指令从公共变量池中查找变量,设置变量的值作为用户指定的基本对象类型所对应的测试数据;

测试用例封装子单元,用于根据多个用例元素及其各基本对象类型对应的测试数据,和已接收的用户输入的用例名称、级别、描述和类型属性,和预期结果值,封装测试用例;

或者,所述测试用例封装单元包括:

数据更新子单元,用于展现用例元素中具有配置信息的基本对象类型,接收用户输入的所述基本对象类型的新的配置信息,更新用例元素;

测试数据获取子单元,用于获取所述用例元素中各基本对象类型对应的测试数据;

测试用例封装子单元,用于根据更新后的用例元素及其各基本对象类型对应的测试数据,封装测试用例。

10. 根据权利要求 1 所述的测试装置,其特征在于,

所述基本对象类型的配置信息包括:基本对象类型的输入值范围;

所述测试用例封装单元,还用于根据已经封装的测试用例和测试用例中基本对象类型的输入值范围,利用二元正交算法生成新的测试用例。

11. 根据权利要求 1 所述的测试装置,其特征在于,还包括:

用例调度单元,用于控制所述接口测试装置展现用例调度配置界面,接收用户在所述用例调度配置界面中指定的测试用例运行规则;

所述用例执行单元,用于根据用户指定的测试用例运行规则,运行所述测试用例。

12. 一种接口测试方法,包括运行测试用例,生成测试报告,其特征在于,所述方法还包括:

根据用户指定的接口参数类型扫描被测试对象;

分析对被测试对象的扫描结果,展现所述接口参数类型的结构,所述接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息;

获取用户选择的用例元素,利用用例元素中基本对象类型的配置信息,获取所述用例元素中各基本对象类型对应的测试数据,利用所述用例元素及其中各基本对象类型对应的测试数据封装测试用例。

13. 根据权利要求 12 所述的方法,其特征在于,

展现所述接口参数类型的结构包括:

当所述接口参数类型为复杂对象类型时,展现所述接口参数类型的树形结构。

14. 根据权利要求 12 所述的方法,其特征在于,

获取用户指定的基本对象类型的配置信息包括:

接收用户输入的所述基本对象类型的配置信息;

或者,

接收用户输入的指示查找所述接口参数类型的配置信息的查找指令;

在本地查找所述接口参数类型的配置信息,如果本地没有找到所述接口参数类型的配

置信息,向中央控制器查询所述接口参数类型的配置信息,其中,所述接口参数类型的配置信息包括:接口参数类型中各基本对象类型的配置信息。

15. 根据权利要求 12 所述的方法,其特征在于,

在获取用户指定的基本对象类型的配置信息之后,该方法还包括:

根据所述基本对象类型的配置信息配置接口参数对象;

封装接口,所述接口包括:用户指定的接口寻址信息、用户指定的接口返回类型所对应的接口返回对象和所述接口参数对象;

所述获取用户选择的用例元素包括:获取用户选择的已封装的接口。

16. 根据权利要求 15 所述的方法,其特征在于,

在配置接口参数对象之后,该方法还包括:

接收到用户指示将接口参数类型的配置信息共享的指令后,保存所述接口参数类型的配置信息以便以后配置接口参数对象时作为用户指定的基本对象类型的配置信息;

或者,

在封装接口之后,该方法还包括:

接收用户指示将接口共享的指令,保存所述接口以便以后封装测试用例时作为用户选择的用例元素;

或者,

在利用所述用例元素及其中各基本对象类型对应的测试数据封装测试用例之后,该方法还包括:

接收用户指示将测试用例共享的指令,保存所述测试用例。

17. 根据权利要求 12 所述的方法,其特征在于,

在获取用户指定的基本对象类型的配置信息之后,该方法还包括:

根据所述基本对象类型的配置信息配置接口参数对象;

封装接口,所述接口包括:用户指定的接口寻址信息、用户指定的接口返回类型所对应的接口返回对象和所述接口参数对象;

获取用户选择的多个逻辑封装元素,按照用户指定的调用关系将多个逻辑封装元素封装成测试逻辑结构,所述多个逻辑封装元素包括所封装的接口;

所述获取用户选择的用例元素包括:获取用户选择的已封装的测试逻辑结构。

18. 根据权利要求 17 所述的方法,其特征在于,

所述多个逻辑封装元素包括第一逻辑封装元素和第二逻辑封装元素;

所述按照用户指定的调用关系将多个逻辑封装元素封装成测试逻辑结构包括:

将第一逻辑封装元素的返回结果作为变量,根据用户的指令设置变量的值作为第二逻辑封装元素的输入值,得到测试逻辑结构。

19. 根据权利要求 17 所述的方法,其特征在于,

在配置接口参数对象之后,该方法还包括:

接收到用户指示将接口参数类型的配置信息共享的指令后,保存所述接口参数类型的配置信息以便以后配置接口参数对象时作为用户指定的基本对象类型的配置信息;

或者,

在封装接口之后,该方法还包括:

接收用户指示将接口共享的指令,保存所述接口以便以后封装测试用例时作为用户选择的用例元素或者封装测试逻辑结构时作为用户选择的逻辑封装元素;

或者,

在封装测试逻辑结构之后,该方法还包括:

接收用户指示将测试逻辑结构共享的指令后,保存所述测试逻辑结构以便以后封装测试用例时作为用户选择的用例元素;

或者,

在利用所述用例元素及其中各基本对象类型对应的测试数据封装测试用例之后,该方法还包括:

接收用户指示将测试用例共享的指令,保存所述测试用例。

20. 根据权利要求 16 或者 19 所述的方法,其特征在于,

所述保存所述接口参数类型的配置信息包括:

在本地保存所述接口参数类型的配置信息,并将所述接口参数类型的配置信息保存到中央控制器。

21. 根据权利要求 16 或者 19 所述的方法,其特征在于,

保存所述接口包括:

在本地保存所述接口,并将所述接口保存到中央控制器。

22. 根据权利要求 19 所述的方法,其特征在于,

保存所述测试逻辑结构包括:

在本地保存所述测试逻辑结构,并将所述测试逻辑结构保存到中央控制器。

23. 根据权利要求 16 或者 19 所述的方法,其特征在于,

保存所述测试用例包括:

在本地保存所述测试用例,并将所述测试用例保存到中央控制器。

24. 根据权利要求 12 所述的方法,其特征在于,

所述获取所述用例元素中各基本对象类型对应的测试数据包括:

展现用例元素中具有配置信息的基本对象类型,根据用户的指令从公共变量池中查找变量,设置变量的值作为用户选择的用例元素中基本对象类型对应的测试数据。

25. 根据权利要求 12 所述的方法,其特征在于,

该方法还包括:控制所述接口测试装置展现用例调度配置界面,接收用户在所述用例调度配置界面中指定的测试用例运行规则;

所述运行所述测试用例包括:

根据用户指定的测试用例运行规则,运行所述测试用例。

接口测试方法及装置

技术领域

[0001] 本发明涉及测试技术领域,特别涉及一种接口测试方法及装置。

背景技术

[0002] 集成测试是在单元测试的基础上,将所有模块按照设计要求组装成系统,对组装的系统进行测试。实践表明,一些模块虽然能够单独地工作,但并不能保证连接起来也能正常的工作,因此集成测试是非常必要的。

[0003] 在软件的单元测试与集成测试中,通常需要对软件的接口进行测试,因为只有保证软件接口的稳定性,才能保证系统的稳定性,所以接口测试称为软件测试中一个重要组成部分。

[0004] 目前业界内出现一些测试框架,比如 JUNIT,其提供了编写接口测试过程中常用的方法,比如结果对比、用例调度等功能。在基于这个框架进行接口测试时,测试人员需要利用 JUNIT 提供的方法,编写测试代码。

[0005] 现有技术的缺点是:

[0006] 现有技术中需要用户利用 JUNIT 框架提供的方法,编写测试代码,测试效率低。

发明内容

[0007] 本发明实施例提供一种接口测试方法及装置,进行接口测试时不需要开发人员编写测试代码,以提高测试效率。

[0008] 有鉴于此,本发明实施例提供:

[0009] 一种接口测试装置,包括:接口扫描单元、数据定义单元、测试用例封装单元、用例执行单元和报告生成单元,

[0010] 所述接口扫描单元,用于根据用户指定的接口参数类型扫描被测试对象;

[0011] 所述数据定义单元,用于分析所述接口扫描单元对所述被测试对象的扫描结果,展现所述接口参数类型的结构,所述接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息;

[0012] 所述测试用例封装单元,用于获取用户选择的用例元素,利用所述数据定义单元获取的用例元素中基本对象类型的配置信息,获取所述用例元素中各基本对象类型对应的测试数据,利用所述用例元素及其中各基本对象类型对应的测试数据封装测试用例;

[0013] 所述用例执行单元,用于运行测试用例封装单元封装的测试用例;

[0014] 所述报告生成单元,用于根据测试用例的运行结果,生成测试报告。

[0015] 一种接口测试方法,包括:

[0016] 根据用户指定的接口参数类型扫描被测试对象;

[0017] 分析对被测试对象的扫描结果,展现所述接口参数类型的结构,所述接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息;

[0018] 获取用户选择的用例元素,利用用例元素中基本对象类型的配置信息,获取所述

用例元素中各基本对象类型对应的测试数据,利用所述用例元素及其中各基本对象类型对应的测试数据封装测试用例;

[0019] 运行所述测试用例,生成测试报告。

[0020] 本发明实施例展现接口参数类型的结构(接口参数类型包括基本对象类型),并获取用户指定的基本对象类型的配置信息,在封装测试用例时,利用基本对象类型的配置信息获取各基本对象类型对应的测试数据,从而封装测试用例,使用户在构造测试用例的过程时不需要写入代码,提高了测试效率。

附图说明

[0021] 为了更清楚地说明本发明实施例的技术方案,下面将对实施例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的附图。

[0022] 图 1 是本发明实施例提供的一接口测试装置结构图;

[0023] 图 2A 是本发明实施例提供的另一接口测试装置结构图;

[0024] 图 2B 是本发明实施例提供的测试用例生成模块 10 部分结构图;

[0025] 图 3A 是本发明实施例提供的寻址信息选择示意图;

[0026] 图 3B 是本发明实施例提供的对象构造界面示意图;

[0027] 图 3C 是本发明实施例提供的对象构造的类图;

[0028] 图 4A 是本发明实施例提供的接口界面示意图;

[0029] 图 4B 是本发明实施例提供的接口封装的类图;

[0030] 图 5A 是本发明实施例提供的逻辑封装界面示意图;

[0031] 图 5B 是本发明实施例提供的逻辑封装的类图;

[0032] 图 6A 是本发明实施例提供的用例封装界面示意图;

[0033] 图 6B 是本发明实施例提供的用例封装的类图;

[0034] 图 7A 是本发明实施例提供的用例调度界面示意图;

[0035] 图 7B 是本发明实施例提供的用例调度的类图;

[0036] 图 8 是本发明另一实施例提供的接口测试装置结构图;

[0037] 图 9A 是本发明一实施例提供的接口测试方法流程图;

[0038] 图 9B 是本发明另一实施例提供的接口测试方法流程图;

[0039] 图 9C 是本发明另一实施例提供的接口测试方法流程图;

[0040] 图 10 是本发明另一实施例提供的接口测试方法流程图。

具体实施方式

[0041] 为使本发明实施例的目的、技术方案和优点更加清楚,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作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。

[0042] 参阅图 1,本发明一实施例提供一种接口测试装置,包括:接口扫描单元 101、数据定义单元 102、测试用例封装单元 105、用例执行单元 201 和报告生成单元 202,

[0043] 接口扫描单元 101,用于根据用户指定的接口参数类型扫描被测测试对象;

[0044] 数据定义单元 102,用于分析接口扫描单元 101 对被测试对象的扫描结果,展现接口参数类型的结构,该接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息;

[0045] 测试用例封装单元 105,用于获取用户选择的用例元素,利用数据定义单元 102 获取的用例元素中基本对象类型的配置信息,获取用例元素中各基本对象类型对应的测试数据,利用用例元素及其中各基本对象类型对应的测试数据封装测试用例;

[0046] 用例执行单元 201,用于运行测试用例封装单元封装的测试用例;

[0047] 报告生成单元 202,用于根据测试用例的运行结果,生成测试报告。

[0048] 其中,接口扫描单元 101、数据定义单元 102 和测试用例封装单元 105 位于测试用例生成模块 10 上;用例执行单元 201 和报告生成单元 202 位于测试用例执行模块 20 上。

[0049] 数据定义单元 102 具体用于分析接口扫描单元 101 对被测试对象的扫描结果,当接口参数类型为复杂对象类型时,展现接口参数类型的树形结构,获取用户指定的接口参数类型中各基本对象类型的配置信息;当接口参数类型为基本对象类型时,获取用户指定的基本对象类型的配置信息。

[0050] 其中,数据定义单元 102 可以通过如下方式获取用户指定的基本对象类型的配置信息:第一种方式为:数据定义单元 102 接收用户输入的基本对象类型的配置信息;第二种方式为:数据定义单元 102 接收用户输入的指示查找接口参数类型的配置信息的查找指令后,从本地的共享单元 106 查找该接口参数类型的配置信息。

[0051] 具体的,数据定义单元 102 还用于在接收到用户指示将接口参数类型的配置信息共享的指令后,将接口参数类型的配置信息保存到共享单元 106;其中接口参数类型的配置信息包括接口参数类型中各基本对象类型的配置信息;测试用例封装单元 105 还用于接收用户指示将测试用例共享的指令,将测试用例保存到共享单元 106;

[0052] 共享单元 106 具体用于保存接口参数类型的配置信息和测试用例;其中,共享单元 106 中保存的接口参数类型中各基本对象类型的配置信息是数据定义单元 102 以后配置接口参数对象时用户所指定的基本对象类型的配置信息。

[0053] 其中,测试用例封装单元 105 中利用数据定义单元 102 获取的用例元素中基本对象类型的配置信息,获取用例元素中各基本对象类型对应的测试数据具体可以通过如下方式实现:展现用例元素中具有配置信息的基本对象类型,根据用户的指令从公共变量池中查找变量,设置变量的值作为用户指定的基本对象类型所对应的测试数据。还可以通过其他方式获取测试数据,具体参见后续实施例对测试用例封装单元 105 的描述。

[0054] 本发明实施例展现接口参数类型的结构(接口参数类型包括基本对象类型),并获取用户指定的基本对象类型的配置信息,在封装测试用例时,利用基本对象类型的配置信息获取各基本对象类型对应的测试数据,从而封装测试用例,使用户在构造测试用例的过程时不需要写入代码,提高了测试效率。

[0055] 参阅图 2A 和 2B,如下对本发明实施例提供的接口测试装置进行详细描述,该接口测试装置可用于软件的集成测试,接口测试装置包括:测试用例生成模块 10 和测试用例执行模块 20,

[0056] 其中测试用例生成模块 10 包括:

[0057] 接口扫描单元 101,用于根据接口定义单元获取的接口参数类型,扫描被测试对象;

[0058] 数据定义单元 102,用于分析接口扫描单元 101 对被测试对象的扫描结果,当接口参数类型为复杂对象类型时,展现接口参数类型的结构,该接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息,根据基本对象类型的配置信息和接口参数类型的结构配置接口参数对象;当接口参数类型为基本对象类型时,获取用户指定的基本对象类型的配置信息,根据基本对象类型的配置信息配置接口参数对象;

[0059] 接口定义单元 103,用于获取用户指定的接口寻址信息、接口返回类型和接口参数类型,封装接口,该接口包括:接口寻址信息,接口返回类型对应的接口返回对象,数据定义单元 102 配置的接口参数对象;

[0060] 测试逻辑封装单元 104,用于获取用户选择的多个逻辑封装元素,按照用户指定的调用关系将多个逻辑封装元素封装成测试逻辑结构,多个逻辑封装元素包括接口定义单元 103 封装的接口;

[0061] 测试用例封装单元 105,用于获取用户选择的用例元素,利用数据定义单元 102 获取的用例元素中基本对象类型的配置信息,获取用例元素中各基本对象类型对应的测试数据,利用用例元素及其中各基本对象类型对应的测试数据封装测试用例;其中,用户选择的用例元素包括测试逻辑封装单元 104 封装的测试逻辑结构;

[0062] 测试用例执行模块 20 包括:

[0063] 用例执行单元 201,用于运行测试用例封装单元封装的测试用例;

[0064] 报告生成单元 202,用于根据测试用例的运行结果,生成测试报告。

[0065] 优选的,该测试用例生成模块 10 还包括共享单元 106 和公共变量池 107;

[0066] 其中,共享单元 106,用于保存数据定义单元 102 配置的接口参数类型的配置信息(接口参数类型的配置信息包括该接口参数类型中基本对象类型的配置信息,基本对象类型的配置信息即为该基本对象类型的各属性值)、接口定义单元 103 封装的接口、测试逻辑封装单元 104 封装的测试逻辑结构和测试用例封装单元 105 封装的测试用例;当在数据定义单元 102 封装接口参数对象时,如果接收到用户选择共享单元 106 中接口参数类型配置信息的指令后,从共享单元 106 获取接口参数类型配置信息以供封装接口的参数对象;当逻辑封装单元 104 封装测试逻辑结构时,如果接收到用户选择共享单元 106 中接口的指令,则从共享单元 106 中获取用户选择的接口;如果接收到用户选择共享单元 106 中测试逻辑结构的指令,从共享单元 106 中获取用户选择的测试逻辑结构,以封装测试逻辑结构;当在测试用例封装单元 105 封装测试用例时,如果接收到用户选择共享单元 106 中接口的指令,从共享单元 106 中获取用户选择的接口;如果接收到用户选择共享单元 106 中测试逻辑结构的指令,从共享单元 106 中获取用户选择的测试逻辑结构,以封装测试用例。优选的,共享单元提供按名称过滤的功能,即根据用户输入的接口参数类型的名称、接口的名称、测试逻辑结构的名称、测试用例的名称查找用户想要的接口参数类型的配置信息、接口、测试逻辑结构和 / 或测试用例,如果在本地没有找到,则向中央控制器查询用户想要的接口参数类型的配置信息、接口、测试逻辑结构和 / 或测试用例。

[0067] 采用上述共享单元 106,当多人同时进行接口测试时,能够通过共享单元 106 从中央控制器直接调用接口参数类型的配置信息、接口、测试逻辑结构和测试用例,避免重复配

置,可以提高接口的测试效率。

[0068] 公共变量池 107,用于保存接口在运行过程中产生的数据,这些数据即为变量,每一个变量都有一个变量名称,规定变量名称以 \$ 字符开始。用户可以在逻辑封装单元 104 中的结果处理中把接口的返回结果保存为一个变量,在用例实际运行时,就会把变量替换成前面接口实际返回的数据,达到不同的接口传递数据的目的。

[0069] 比如 A 接口返回一个结果数据 result,可以在 A 接口的“结果处理”中调用公共变量池的方法保存该数据为一个变量,变量名称为 \$result,如果在 B 接口中某个参数就想输入 A 接口的返回结果,那么这就可以输入 \$result 变量。在实际运行过程中,运行到 B 接口,就会把 A 接口的返回值用于 B 接口。这就是变量的作用。在实际应用中,可以把公共变量池提供给用户进行操作,用户可以在界面上查看公共变量池中所有的变量,并可直接往公共变量池注入一个变量,这些变量可以在任何输入数据的地方使用。

[0070] 如下对测试生成装置 10 中的各单元的功能进行详细介绍:

[0071] 1) 如下具体描述接口扫描单元 101、数据定义单元 102 和接口定义单元 103 的功能:

[0072] 接口扫描单元 101,接收用户输入的被测试信息,根据被测试信息获取待测试接口信息,展现包含待测试接口信息的界面。

[0073] 具体的,如果用户要测试的是本地接口,则用户输入的被测试信息即为被测试对象,其可以是 jar 包,接口扫描单元 101 控制接口测试装置展现该被测试对象包含的接口信息;如果用户要测试远程接口,则用户输入的被测试信息为远程服务器的 IP 地址,接口扫描单元 101 通过该 IP 地址连接到远程服务器,得到该服务器上所有的服务名称相关信息,其中包括服务名称的寻址信息,例如 server1:001。接口扫描单元 101 然后根据该 IP 地址找到用于该远程服务器接口测试的被测试对象,该被测试对象可以是一 jar 包,展现接口信息及服务名称相关信息,其中,接口信息包括各接口的参数类型。

[0074] 其中,上述接口参数类型包括接口参数类型对应的包的路径及其对应的类的名称。

[0075] 接口定义单元 103 包括:接口子单元 1031、第一封装子单元 1032 和第二执行存储单元 1033;接口定义单元 103 中的接口单元 1031 获取用户根据待测试接口信息所指定的接口寻址信息、接口返回类型和接口参数类型,通知接口扫描单元 101 扫描该接口参数类型的结构,其中,用户输入的接口参数类型是接口扫描单元 101 所展现的接口参数类型。

[0076] 由于在接口测试对“本地接口”调用时,实际上在代码中就是要调用某个类的某个接口,例如:QQ 类的 login 接口;其中,接口的寻址信息就是如何调用接口的信息,图 3A 为提供用户选定接口寻址信息的界面。

[0077] 第一种方式,用户选择了静态方法,由于静态方法不需要实例化类,系统默认就是用 QQ 作为寻址信息,所以不需要用户输入任何值。

[0078] 第二种方式,用户选择了单例,这种情况是类 QQ 本身要求是单例(即整个系统只能有一个实例),编码是这样的,QQ qq=QQ.getInstance(). 所以采用这种方式,需要 QQ 提供静态的方法 getInstance(),获取其 QQ 实例。这种方式也不需要用户输入任何值。

[0079] 第三种方式,用户选择了无参数构造,这种情况是类 QQ 类提供了无输入参数的构造函数,编码是这样的 QQ qq=new QQ(),这种方式无需要用户输入任何值。

[0080] 第四种方式,用户选择了自定义的方式,用户需要写入一段脚本,例如编码方式是:QQ qq=new QQ(“ip”,“port”),让用户可以输入 request=new QQ(“ip”,“port”),其中 QQ 需要写完整的包路径。或者如果编码的方式为:QQ qq=QQFactory.getQQ(),则用户输入 request=new QQFactory.getQQ(),其中 QQFactory 需要写上完整的包路径。

[0081] 对于远程接口,用户根据展现的多个服务名称相关信息,选择服务名称,并输入该服务名称对应的命令码,其中,服务名称及其对应的命令码组成接口的寻址信息。

[0082] 接口扫描单元 101 扫描接口参数类型的结构,具体的,使用反射的技术得到该接口中的方法以及方法的参数类型等等。具体的,对于本地接口,扫描用户输入的被测试对象中的接口参数类型的结构;对于远程接口,扫描远程服务器的 IP 地址所对应的 jar 包中的接口参数类型的结构。

[0083] 数据定义单元 102 包括:接口参数对象配置子单元 1021 和第一执行存储单元 1022,其中接口参数对象配置子单元 1021 分析接口扫描单元 101 的扫描结果,当接口参数类型为复杂对象类型时,展现接口参数类型的结构,该接口参数类型包括基本对象类型,接收用户输入的基本对象类型的配置信息(基本对象类型的配置信息为用户输入的基本对象类型的各属性值),或者接收用户输入的指示查找接口参数类型的配置信息的查找指令,从本地共享单元获取该接口参数类型的配置信息(接口参数类型的配置信息包括接口参数类型中基本对象类型的配置信息,基本对象类型的配置信息包括基本对象类型的各属性值),根据基本对象类型的配置信息和接口参数类型的结构配置接口参数对象;当接口参数类型为基本对象类型时,接收用户输入的基本对象类型的配置信息(基本对象类型的配置信息为用户输入的基本对象类型的各属性值),或者接收用户输入的指示查找接口参数类型的配置信息的查找指令,从本地共享单元 106 获取该接口参数类型的配置信息(接口参数类型的配置信息包括接口参数类型中基本对象类型的配置信息,基本对象类型的配置信息包括基本对象类型的各属性值),根据基本对象类型的配置信息配置接口参数对象。第一执行存储单元 1022 用于在接收到用户指示将接口参数对象的配置信息共享的指令后,将接口参数对象的配置信息保存到共享单元 106,以便后续封装接口参数对象时供用户使用。

[0084] 具体的,当接口参数类型为复杂对象类型时,数据定义单元 102 控制接口测试装置以树的形式展现该接口参数类型的结构,如图 3A 所示。

[0085] 其中,基本对象类型的属性包括业务名称和默认值,或者基本对象类型的属性包括业务名称和输入值范围,基本对象类型的属性还可以包括:注释等。具体的,业务名称是该基本对象在接口调用中表示的业务含义,其可以是 Username, Password 等信息,在后续封装测试用例时仅显示具有业务名称的基本对象类型,屏蔽接口参数对象的结构特征,以便方便测试用例的组装和后续的维护;默认值是在基本对象类型没有输入值的情况下的取值;注释是用户输入的其他相关信息。

[0086] 当接口参数类型为复杂对象类型时,图 3B 示出了该接口参数类型的结构,其以树的方式展现了接口参数类型的结构。当用户需要为某个基本对象类型的属性赋值时,单击该基本对象类型,则数据定义单元 102 会控制接口测试装置展示该基本对象类型的属性表,用户在表中填入每个属性的数值,数据定义单元 102 接收用户输入的基本对象类型属性的数值。图 3B 对应的采用 Java 语言实现的类图如图 3C 所示,其中,IDataNode:为 Interface,表示其他类的基类;ComplexNode,表示复杂对象类型,其包含

多个 ConstructNode, 多个 MethodNode, 多个成员对象类型, 其中成员对象类型可以是 SimpleNode 和 ComplexNode, 另外包含一些其他属性 ;ConstructNode :表示构造函数的节点, 包含构造函数的名称, 参数类型, 其中参数类型可以是 SimpleNode 和 ComplexNode ; MethodNode :表示普通方法的节点 ;包含方法名称, 方法的返回类型, 参数类型, 其中参数类型可以是 SimpleNode 和 ComplexNode。

[0087] 需要说明的是, 当接口参考类型为复杂对象类型时, 如果该接口参数类型为抽象类或者 interface 类, 而需要用户将该接口参数类型改为可实例化的子类, 例如, 用户点击接口参数类型(比如图 3A 中作为树根的复杂对象), 数据定义单元 102 控制接口测试装置弹出一个类型更改界面, 用户在该界面上将该接口参数类型改为可实例化的子类。

[0088] 如果用户需要复制某个普通方法, 例如用户需要复制普通方法 1, 则用户在普通方法 1 上点击右键, 则数据定义单元 102 控制接口测试装置弹出该方法的属性菜单, 则用户在该菜单上选择“copy”, 接口测试装置展现一个相同的普通方法 1, 则图 3A 中会出现两个普通方法 1。

[0089] 如果出现循环递归的情况, 比如成员变量的叶子节点中包括自己, 则数据定义单元 102 会提示出错。

[0090] 接口定义单元 103 中的第一封装子单元 1032 从数据定义单元 102 获取接口参数对象, 封装接口, 展现接口界面供用户查看, 该接口包括 :接口寻址信息, 接口返回类型对应的接口返回对象。第二执行存储单元 1033 接收到用户用户指示将接口参数类型的配置信息共享的指令后, 将接口参数类型的配置信息保存到共享单元, 以便以后封装接口参数对象时供用户使用 ;接口参数类型的配置信息包括接口参数类型中基本对象类型的配置信息。

[0091] 其中, 接口定义单元 103 提供的接口界面图如图 4A 所示, 其对应的采用 Java 语言实现的类图如图 4B 所示。其中, IDataNode :为 Interface, 表示所有对象的基类, 这里可以是 ComplexNode 或者 SimpleNode ;InterfaceNode, 表示接口类型的节点, 其包含多个参数对象, 返回值, 另外包含接口的寻址信息和其他的属性。

[0092] 用户可以通过接口界面重新为该接口参数对象中基本对象类型的属性赋值, 比如可以修改业务名称、默认值、输入值范围和 / 或注释。具体操作是 :用户点击接口参数对象, 接口定义单元 103 展现该接口参数对象的结构, 用户点击某个基本对象类型, 修改该基本对象类型的业务名称、接口调用时的默认值、接口调用时输入值范围和 / 或注释。

[0093] 2)、如下是测试逻辑封装单元 104 功能的详细介绍 :

[0094] 逻辑封装单元 104 包括 :第一获取子单元 1041, 第二获取子单元 1042, 第二封装子单元 1043、后置处理方法提供子单元 1044、第三执行存储单元 1045, 其中,

[0095] 第一获取子单元 1041, 用于接收用户选择接口定义单元 102 封装的接口的指令, 从接口定义单元获取接口 ;

[0096] 第二获取子单元 1042, 用于接收用户选择共享单元 106 中接口的指令, 从共享单元中 106 获取用户选择的接口 ;接收用户选择共享单元 106 中测试逻辑结构的指令, 从共享单元 106 中获取用户选择的测试逻辑结构 ;

[0097] 第二封装子单元 1043, 用于接收用户指定的多个逻辑封装元素间的调用关系, 按照该调用关系将多个逻辑封装元素组成测试逻辑结构 ;其中, 多个逻辑封装元素包括 :从

接口定义单元获取的接口、从共享单元中获取的接口和 / 或从共享单元中获取的测试逻辑结构。

[0098] 假定上述多个逻辑封装元素包括第一逻辑封装元素和第二逻辑封装元素,具体的,第二封装子单元 1043 提供逻辑封装界面,根据用户的指令将第一逻辑封装元素的返回结果保存为一个变量,根据用户的指令设置变量的值作为第二逻辑封装元素的输入值,得到测试逻辑结构。

[0099] 需要说明的是,用户可以通过逻辑关系运算符中的开始符号、过程符号和结束符号指定逻辑封装元素间的调用顺序,比如用户指定接口一、接口二、接口三顺序调用。

[0100] 用户可以通过逻辑关系运算符中的定时符号指定某个逻辑封装元素的执行时间;

[0101] 用户可以通过逻辑关系运算符中的循环符号指定某个逻辑封装元素循环执行,并指定该逻辑封装元素循环指定的次数等。

[0102] 假定多个逻辑封装元素包括第一逻辑封装元素、第二逻辑封装元素和第三逻辑封装元素,具体的,第二封装子单元 1043 根据用户的指令将第一逻辑封装元素的返回结果保存为一个变量,利用分支符号及变量,将第一逻辑封装元素分别与第二逻辑元素和第三逻辑元素连接,具体的,可以设置变量值为真(1)时执行第二逻辑封装元素,变量值为假(0)时执行第三逻辑封装元素。

[0103] 后置处理方法提供子单元 1044,用于提供接口调用完成后的后置处理方法供用户选择,比如 getResult(),可以得到接口返回的结果;getParameter(int index),得到接口调用时的参数实例;public_put(String key, Object Value),往公共变量池加入变量;for(i in array) 的循环语句;if(xxx) 分支语句;测试逻辑封装单元 104 并提供类似于 Junit 框架中的 Assert 中的所有对比方法等等,这些方法可以让用户自己扩展,后置处理方法提供子单元提供编辑提示,方便用户输入。

[0104] 第三执行存储单元 1045,用于接收用户指示将测试逻辑结构共享的指令,将测试逻辑结构保存到共享单元,以便以后封装测试用例时供使用使用。

[0105] 其中,逻辑封装界面可以如图 5A 所示,其对应的采用 Java 语言实现的类图如图 5B 所示,其中,IDataNode:为 Interface,表示简单对象和复杂对象,即 ComplexNode 或者 SimpleNode。LogicNode,表示所有逻辑控制节点的基类。StartLogicNode:逻辑开始节点,该节点为一个逻辑封装的入口,框架会为其分配一个唯一的 ID 进行标示,该节点只允许添加一个孩子节点。InterfaceExecuteNode,接口执行节点,该节点可以添加一个或者多个接口,每个接口按顺序执行,该接口只能添加一个孩子节点。

[0106] 3)、如下是测试用例封装单元 105 功能的详细介绍:

[0107] 测试用例封装单元 105 包括第三获取子单元 1051、第四获取子单元 1052、第五获取子单元 1053、测试数据获取子单元 1054、预期结果赋值单元 1055、测试用例封装子单元 1056、第四执行存储单元 1057,

[0108] 第三获取子单元 1051,用于接收用户选择测试逻辑封装单元 104 封装的测试逻辑结构的指令,从测试逻辑封装单元 104 获取测试逻辑结构;

[0109] 第四获取子单元 1052,用于接收用户选择接口定义单元封装的接口的指令,从接口定义单元获取接口;

[0110] 第五获取子单元 1053,用于接收用户选择共享单元 106 中接口的指令,从共享单元 106 中获取用户选择的接口;接收用户选择共享单元 106 中测试逻辑结构的指令,从共享单元 106 中获取用户选择的测试逻辑结构;

[0111] 测试数据获取子单元 1054,用于获取多个用例元素中基本对象类型对应的测试数据;其中,多个用例元素包括从测试逻辑封装单元获取的测试逻辑结构、从共享单元中获取的接口和/或从共享单元中获取的测试逻辑结构。

[0112] 具体的获取测试数据的方式包括但不限于如下几种方式:

[0113] 第一种方式:设置用户从公共变量池中选定的变量的值作为用例元素中基本对象类型对应的测试数据。具体的,用户从公共变量池中选定变量,针对该基本对象类型输入以\$开始的变量名称,测试用例封装单元 105 接收到用户输入的变量名称后,设置该变量的值为基本对象类型对应的测试数据。

[0114] 第二种方式:接收用户直接输入的具体数值为测试数据。

[0115] 第三种方式:测试数据获取子单元提供常用赋值方法供用户选择,常用方法可以是:得到随机的数字的方法:getRandNum();字符串转 byte 数据的方法,Str2byte(String str)等等。

[0116] 如果用户没有进行赋值,则测试数据获取子单元 1054 采用默认值为测试数据。

[0117] 预期结果赋值单元 1055,用于根据用户的指令设置测试用例的预期结果值。

[0118] 测试用例封装子单元 1056,用于根据多个用例元素(此处多个用例元素为一个用例实体)及其中各基本对象类型对应的测试数据,和已接收的用户输入的用例名称、级别、描述和类型属性,和预期结果值,封装测试用例。其中用例名称、级别、描述、类型属性主要为用户在维护过程中使用,方便阅读接口完成的功能。用例级别与用例类型可以在用例调度中使用(按用例级别运行,按用例类型运行)。

[0119] 第四执行存储单元 1057,用于接收用户指示将测试用例共享的指令后,将测试用例保存到共享单元中。

[0120] 或者,所述测试用例封装单元包括:

[0121] 数据更新子单元,用于展现用例元素中具有配置信息的基本对象类型,接收用户输入的所述基本对象类型的新的配置信息,更新用例元素;

[0122] 测试数据获取子单元,用于获取所述用例元素中各基本对象类型对应的测试数据;

[0123] 测试用例封装子单元,用于根据更新后的用例元素及其各基本对象类型对应的测试数据,封装测试用例。

[0124] 测试用例封装单元 105 还包括:基本对象信息更新子单元;

[0125] 基本对象信息更新子单元,用于接收用户对用于组成用例实体的元素中基本对象类型的属性值修改的指令,保存用户为基本对象类型的属性赋予的新值。实现可以在封装测试用例时修改基本对象类型属性值的目的。

[0126] 测试用例封装单元 105 还包括:自动生成用例子单元;

[0127] 自动生成用例子单元,用于根据已经封装的测试用例和测试用例中基本对象类型的输入值范围,使用二元正交算法生成测试用例。实现可以自动生成用例的目的。

[0128] 例如,假定有已配置好一个测试用例,该测试用例是:买东西,给钱,找钱;

[0129] 这里的参数有 : 什么东西, 买多少, 给多少钱 ;

[0130] 预期是找回多少钱 ;

[0131] 其接口调用时能使用的输入值为 :

[0132] 什么东西 : 苹果、雪梨、电视机

[0133] 买多少 : 1、2、1000

[0134] 给多少钱 : 10、20、100

[0135] 使用自动生成用例功能, 就会自动组合生成以下的用例 :

[0136] 买 1 个苹果, 给 20 块钱, 预期找回多少

[0137] 买 1000 台电视机, 给 100 块, 预期被告诉不够钱, 等等

[0138] 测试用例包括两类节点 : 特性节点和用例节点 (即上述封装的测试用例), 其中特性节点是包含一个或者多个用例节点和特性节点。测试用例封装单元 105 还可以根据用户的指示将多个测试用例划到一个特性节点, 以便后续在维护时方便用户操作。

[0139] 测试用例封装单元 105 控制接口测试装置展现的用例封装界面可以如图 6A 所示, 假定用户在该界面上选择的树根用例为一个特性节点, 测试用例封装单元 105 把“特性属性”页签激活, 并以表的方式展现该特性节点的直接孩子节点, 比如其包含特性节点 1, 特性节点 2, 用例节点 3, 并在表的下方显示统计信息 (有多少个特性节点, 多少个用例节点); 当用户点击特性节点 2, 测试用例封装单元 105 把“特性属性”页签激活, 并以表的展示特性节点 2 的直接孩子节点, 比如其包含用例节点 1 和用例节点 2, 并在表的下方显示统计信息; 用户点击用例节点 1, 测试用例封装单元 105 则把“用例属性”页签激活, 并展示用例属性, 包括用例名称, 用例级别, 用例描述, 用例类型, 其中这些信息可接受用户修改。用户可以点击“用例实体”页签, 测试用例封装单元 105 以表的方式展示用例需要调用的用例元素, 如接口和 / 或测试逻辑结构, 当用户选择了一个具体的测试逻辑结构, 假定用户点击“登陆”哪一行, 则在图中下方的“用例操作区域”的“参数赋值”页签就会激活, 用户就可以输入用例的输入值了。用户也可以点击“结果处理”, 则用户可以设定接口测试的预期结果值, 或者保存一些结果到公共变量池等处理动作。

[0140] 需要说明的是, 这里不会展示接口参数对象的树型结构, 这里使用扁平化的方式展示, 即使用表展示接口参数对象中业务名称不为空的基本对象, 展示的字段信息有 : 业务名称, 类型, 默认值, 输入值, 输入值范围, 注释。这里不显示接口参数对象结构的好处是把不相关的信息屏蔽掉, 方便用户操作, 用户可以直接在“输入值”的字段完成用例的测试数据输入。同时这里也提供入口跳转到数据定义单元查看接口参数对象的详细信息, 用户只需要点击右键即可弹出菜单, 选择“location”即可跳转。

[0141] 其中, 图 6A 对应的采用 Java 语言实现的类图如图 6B 所示, 其中, ICaseNode : 为 Interface, 表示用例节点与特性节点的基类 ; StartLogicNode, 表示所有逻辑封装的开始节点 ; UserCaseNode : 表示用例节点, 该节点包含一个或者多个逻辑封装, 即 StartLogicNode ; FeatureNode, 表示特性节点, 该节点包含多个 ICaseNode, 即特性 FeatureNode 和 UserCaseNode。

[0142] 优选的, 该接口测试装置可以连接到一台独立的服务器上, 在该服务器上共享自己已有的配置 (比如共享单元中的内容), 以便其他接口测试装置使用。

[0143] 本发明实施例采用第三方组件 Hibernate 保存模型数据到数据库中, 当接口测试

装置重新启动时,先从数据库获取数据,建立数据模型,然后再在界面上显示,方便用户操作,这样用户就不用在重新启动时执行重新输入被测试信息供接口扫描单元进行扫描等操作了。

[0144] 优选的,测试用例执行模块 20 还包括:用例调度单元 203,其中,

[0145] 该用例调度单元可以提供多个测试用例供用户选择,将用户选择的多个测试用例组成测试实例,后续用例执行单元 201 会执行测试实例;优选的,该用例调度单元还用于控制接口测试装置展现用例调度界面,接收用户在用例调度界面中指定的测试实例的运行规则;用例执行单元 201 将根据用户指定的测试实例的运行规则,运行测试实例。其中,测试实例的运行规则即为该测试实例中每个测试用例的运行规则。

[0146] 具体的,用例调度单元将用户指定的测试实例的运行规则导出到 xml 文件中,用例执行单元 201 加载 xml 文件中的测试实例的运行规则,然后按照测试用例的运行规则运行测试用例。

[0147] 测试实例的运行规则包括但不限于如下几种规则:

[0148] 循环运行策略:每个测试用例连续运行 x 次,并暂停 y 秒,然后所有测试用例重复运行 z 次;

[0149] 定时运行策略:在 xx:xx 开始运行测试实例;

[0150] 用例筛选策略:用例特性名称、用例名称、用例级别,用例类型满足条件,该测试用例才能运行。

[0151] 其中,用例调度界面可以使用如图 7A 所示,用户可以选择一些测试用例并设定测试用例的运行规则,组成了一个测试实例(如图中左边的树形结构)。用户可以组织多个测试实例供不同的场景使用。图 7A 对应的采用 Java 语言实现的类图如图 7B 所示。其中, AInvoker:为抽象类,表示用所有调度器的基类;UserCaseNode,表示用例节点;UserCaseNodeInvoker:表示调用一个或者多个测试用例的调度器,即测试用例的运行规则;LevelInvoker:表示按级别的调度器;TimeInvoker:表示定时启动的调度器;RepeatStrategyInvoker:表示重复调用的调器。其中,除了 UserCaseNodeInvoker 调度器是只能执行测试用例外,其他的各个调度器均可以嵌套使用(比如,可以定时启动一个重复执行测试用例的调度器)。如果需要扩展调度器,只需要继承 AInvoker 即可。

[0152] 测试用例执行单元 201 的功能具体包括如下:

[0153] 测试用例执行单元 201 具体根据测试数据(即测试用例封装单元获取的用户指定的测试数据)及测试逻辑结构(或接口),生成脚本语言(JavaScript 或者 Python),然后使用 Java 的脚本引擎执行脚本语言得到接口的参数实例;根据接口的寻址信息,使用 java 的反射技术调用接口,然后生成结果判断脚本(JavaScript 或者 Python),使用 Java 的脚本引擎执行结果判断脚本来判断接口测试是否成功,具体的,把接口执行结果传入到脚本引擎,然后使用脚本引擎执行结果判断脚本,验证接口测试是否成功。

[0154] 本发明实施例在接口参数类型为复杂对象类型时,自动解析并展现该接口参数类型的结构,使用户在构造接口的参数对象时不需要写入代码,而仅在对象构造界面输入基本对象类型的各属性值就可以了,该测试装置同时提供逻辑封装界面供用户指定多个接口之间的调用关系,使在集成测试环境中,不需要用户写入任何代码,而是仅需要用户根据界面的提示信息进行选择,实现集成测试场景中对多个接口进行测试;同时在用户增加/删

除 / 修改用例时都不需要变更测试代码,方便用户操作,节省接口测试的时间开销;进一步,采用图像化的方式供用户配置测试用例及测试数据,方便用户操作,从而可以实现任何系统的集成测试;进一步,共享单元可以共享接口的参数对象、接口、测试逻辑结构、用例等,方便多人同时进行接口测试。

[0155] 参阅图 8,本发明另一实施例提供一种接口测试装置,其与上述实施例的区别在于,测试用例生成模块 10 中不包括测试逻辑封装单元 104,该接口测试装置可用于软件的单元测试,如下简单介绍该接口测试装置中的部分单元功能:

[0156] 接口扫描单元 101,用于根据接口定义单元获取的接口参数类型,扫描被测试对象;

[0157] 数据定义单元 102,用于分析接口扫描单元对被测试对象的扫描结果,当接口参数类型为复杂对象类型时,展现接口参数类型的结构,该接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息,根据基本对象类型的配置信息和接口参数类型的结构配置接口参数对象;当接口参数类型为基本对象类型时,获取用户指定的基本对象类型的配置信息,根据基本对象类型的配置信息配置接口参数对象;

[0158] 接口定义单元 103,用于获取用户指定的接口寻址信息、接口返回类型和接口参数类型,封装接口,该接口包括:接口寻址信息,接口返回类型对应的接口返回对象,数据定义单元 102 配置的接口参数对象;

[0159] 测试用例封装单元 105,用于获取用户选择的用例元素,利用用例元素中基本对象类型的配置信息获取用例元素中各基本对象类型对应的测试数据,利用用例元素及其中各基本对象类型对应的测试数据封装测试用例;其中,用户选择的用例元素包括接口定义单元封装的接口;

[0160] 用例执行单元 201,用于运行测试用例封装单元封装的测试用例;

[0161] 报告生成单元 202,用于根据测试用例的运行结果,生成测试报告。

[0162] 其各单元的功能与前述实施例相似,在此不再赘述。该实施例展现接口参数类型的结构(接口参数类型包括基本对象类型),并获取用户指定的基本对象类型的配置信息,并利用基本对象类型的配置信息获取各基本对象类型对应的测试数据,从而封装测试用例,使用户在构造测试用例的过程时不需要写入代码,提高了测试效率。

[0163] 对于上述实施例中的数据定义单元、接口定义单元、测试逻辑封装单元、测试用例封装单元以及共享单元和公共变量池,根据用户的输入信息,均可以建立对应的数据模型,然后将模型数据保存到数据库中,当接口测试装置重新启动后,先从数据库获取模型数据,这些模型数据通过转换器,使用 GUI 的方式展示给用户,方便用户操作。

[0164] 参阅图 9A,本发明一实施例提供一种接口测试方法,该方法具体包括:

[0165] 901A、根据用户指定的接口参数类型扫描被测试对象;

[0166] 该步骤可以由接口扫描单元执行。

[0167] 902A、分析对被测试对象的扫描结果,展现接口参数类型的结构,其中,该接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息;

[0168] 该步骤可以由数据定义单元执行。具体的,数据定义单元可以通过如下方式获取用户指定的基本对象类型的配置信息:第一种方式为:数据定义单元接收用户输入的基本对象类型的配置信息;第二种方式为:数据定义单元接收用户输入的指示查找接口参数类

型的配置信息的查找指令后,从本地的共享单元查找该接口参数类型的配置信息

[0169] 903A、获取用户选择的用例元素,利用用例元素中基本对象类型的配置信息,获取用例元素中各基本对象类型对应的测试数据,利用用例元素及其中各基本对象类型对应的测试数据封装测试用例;

[0170] 该步骤可以由测试用例封装单元执行,其中,获取用例元素中各基本对象类型对应的测试数据具体可以通过如下方式实现:展现用例元素中具有配置信息的基本对象类型,根据用户的指令从公共变量池中查找变量,设置变量的值作为用户指定的基本对象类型所对应的测试数据。还可以通过其他方式获取测试数据,具体参见上述装置实施例中对测试用例封装单元的描述。

[0171] 904A、运行测试用例,生成测试报告。

[0172] 本发明实施例展现接口参数类型的结构(接口参数类型包括基本对象类型),并获取用户指定的基本对象类型的配置信息,在封装测试用例时,利用基本对象类型的配置信息获取各基本对象类型对应的测试数据,从而封装测试用例,使用户在构造测试用例的过程时不需要写入代码,提高了测试效率。

[0173] 参阅图 9B,本发明另一实施例提供一种接口测试方法,该方法可适用于软件的单元测试,该方法具体包括:

[0174] 901B、接口测试装置获取用户指定的接口寻址信息、接口返回类型和接口参数类型;

[0175] 该步骤可以由接口定义单元执行。

[0176] 在该步骤之前,接口测试装置中的接口扫描单元会接收用户输入的被测试信息,根据被测试信息获取待测试接口信息,展现包含待测试接口信息的界面,然后接口定义单元接收用户根据待测试接口信息所输入的接口寻址信息、接口返回类型和接口参数类型,并通知接口扫描单元 101 扫描该接口参数类型的结构。

[0177] 其中,待测试接口可以是本地接口也可以是远程接口,接口测试装置针对不同的接口类型所进行的处理方式见前面装置部分的描述,在此不再赘述。

[0178] 902B、接口测试装置根据接口参数类型,扫描被测试对象。

[0179] 该步骤可以由接口扫描单元执行。

[0180] 对于本地接口,该步骤是扫描用户输入的被测试对象中的接口参数类型的结构;对于远程接口,该步骤是扫描远程服务器的 IP 地址所对应的 jar 包中的接口参数类型的结构。

[0181] 903B、分析接口对被测试对象的扫描结果,当接口参数类型为复杂对象类型时,展现接口参数类型的结构,该接口参数类型包括基本对象类型,获取用户指定的基本对象类型的配置信息,根据基本对象类型的配置信息和接口参数类型的结构配置接口参数对象;当接口参数类型为基本对象类型时,获取用户指定的基本对象类型的配置信息,根据基本对象类型的配置信息配置接口参数对象。

[0182] 该步骤可以由数据定义单元执行。

[0183] 其中,基本对象类型的属性包括:业务名称、默认值、输入值范围。

[0184] 其中,获取用户指定的基本对象类型的配置信息具体可以有如下实现方式:第一种方式为:接收用户输入的基本对象类型的配置信息;第二种方式为:接收用户输入的指

示查找接口参数类型的配置信息的查找指令；在本地查找该接口参数类型的配置信息，如果本地没有找到该接口参数类型的配置信息，向中央控制器查询该接口参数类型的配置信息，其中，该接口参数类型的配置信息包括：接口参数类型的基本对象类型的配置信息。

[0185] 在配置接口参数对象之后，该步骤还可以接收用户输入的指示将接口参数对象的配置信息共享的指令后，将接口参数对象的配置信息保存到共享单元以便以后配置接口参数对象时供用户使用。

[0186] 904B、接口测试装置封装接口，该接口包括：接口寻址信息，接口返回类型对应的接口返回对象，所配置的接口参数对象。

[0187] 该步骤可以由接口定义单元执行。该步骤还可以接收用户指示将接口共享的指令，保存接口以便以后封装测试用例时作为用户选择的用例元素。

[0188] 905B、获取用户选择的用例元素，利用用例元素中基本对象类型的配置信息，获取用例元素中各基本对象类型对应的测试数据，利用用例元素及其中各基本对象类型对应的测试数据封装测试用例；其中，用户选择的用例元素包括接口。

[0189] 该步骤可以由测试用例封装单元执行。

[0190] 该步骤中还可以接收用户指示将测试用例共享的指令，保存测试用例到共享单元中。

[0191] 906B、运行测试用例，生成测试报告。

[0192] 该步骤由用例调度单元、用例执行单元和报告生成单元执行。

[0193] 具体的，该接口测试装置中的用例调度单元提供多个测试用例供用户选择，将用户选择的多个测试用例组成测试实例，并控制接口测试装置展现用例调度界面，接收用户在用例调度界面中指定的测试实例的运行规则，接口测试装置中的用例执行单元将根据用户指定的测试实例的运行规则，运行测试实例。其中，测试实例的运行规则即为该测试实例中每个测试用例的运行规则。

[0194] 本发明实施例在接口参数类型为复杂对象类型时，自动解析并展现该接口参数类型的结构（该接口参数类型包括基本对象类型），并获取用户指定的基本对象类型的配置信息，使用户在构造接口参数对象时不需要写入代码，并利用基本对象类型的配置信息获取各基本对象类型对应的测试数据，从而封装测试用例，使用户在构造测试用例的过程时都不需要写入代码，提高了测试效率。

[0195] 参阅图 9C，本发明另一实施例提供一种接口测试方法，该方法可适用于软件的集成测试，该方法具体包括：

[0196] 901C-904C 与 901B-904B 相同，在此不再赘述。

[0197] 905C、获取用户选择的多个逻辑封装元素，按照用户指定的调用关系将多个逻辑封装元素封装成测试逻辑结构，多个逻辑封装元素包括所封装的接口。

[0198] 该步骤可以由测试逻辑封装单元执行。

[0199] 具体的，接口测试装置提供逻辑封装界面，根据用户的指令将第一逻辑封装元素的返回结果保存为一个变量，利用该变量和逻辑关系运算符将第一逻辑封装元素与第二逻辑封装元素连接，将多个逻辑封装元素封装成一个测试逻辑结构，其中，第一逻辑封装元素和第二逻辑封装元素可以是已封装的接口或者测试封装结构。

[0200] 906C、获取用户选择的用例元素，利用用例元素中基本对象类型的配置信息，获取

用例元素中各基本对象类型对应的测试数据,利用用例元素及其中各基本对象类型对应的测试数据封装测试用例;其中用户选择的用例元素包括测试逻辑结构。

[0201] 该步骤可以由测试用例封装单元执行。

[0202] 其中,可以有三种测试数据获取方式,具体见前面装置部分的相应描述。

[0203] 该步骤具体是根据多个用于组成用例实体的元素及其中各基本对象对应的测试数据,和已接收的用户输入的用例名称、级别、描述和类型属性,和预期结果值,封装测试用例。其中用例名称、级别、描述、类型属性主要为用户在维护过程中使用,方便阅读接口完成的功能。用例级别与用例类型可以在用例调度中使用(按用例级别运行,按用例类型运行),用例实体可以包含一个或者多个测试逻辑结构或者接口。

[0204] 907C、接口测试装置运行测试用例,生成测试报告。

[0205] 该步骤由用例调度单元、用例执行单元和报告生成单元执行。

[0206] 具体的,该接口测试装置中的用例调度单元提供多个测试用例供用户选择,将用户选择的多个测试用例组成测试实例,并控制接口测试装置展现用例调度界面,接收用户在使用调度界面中指定的测试实例的运行规则,接口测试装置中的用例执行单元将根据用户指定的测试实例的运行规则,运行测试实例。其中,测试实例的运行规则即为该测试实例中每个测试用例的运行规则。

[0207] 本发明实施例在接口参数类型为复杂对象类型时,自动解析并展现该接口参数类型的结构(该接口参数类型包括基本对象类型),并获取用户指定的基本对象类型的配置信息,使用户在构造接口的参数对象时不需要写入代码,该接口测试装置同时根据用户指定的多个接口之间的调用关系将多个接口封装成测试逻辑结构,使在集成测试环境中,不需要用户写入任何代码,而是仅需要用户根据界面的提示信息进行选择,实现集成测试场景中对多个接口进行测试;同时在用户增加/删除/修改用例时都不需要变更测试代码,方便用户操作,节省接口测试的时间开销。进一步,采用图像化的方式供用户配置测试用例及测试数据,方便用户操作,从而可以实现任何系统的集成测试。

[0208] 参阅图 10,本发明实施例提供一种接口测试方法,该方法具体包括:

[0209] 1001、接口测试装置接收用户输入的被测试信息,根据被测试信息获取待测试接口信息,展现包含待测试接口信息的界面。

[0210] 1002、接口测试装置获取用户指定的接口寻址信息,接口参数类型及接口返回类型。其中,用户是根据待测试接口信息指定接口寻址信息,接口参数类型及接口返回类型的。

[0211] 1003、根据接口参数类型,扫描被测试对象,分析接口对被测试对象的扫描结果,当接口参数类型为复杂对象类型时,展现接口参数类型的结构,该接口参数类型的结构由基本对象类型组成,接收用户输入的指示查找该接口参数类型的配置信息的查找指令,查找本地的共享单元中是否有该接口参数类型的配置信息,如果否,执行 1004,如果是,执行 1009。

[0212] 1004、到中央控制器查找该接口参数类型的配置信息,如果没有找到,执行 1005,如果找到,执行 1008。

[0213] 1005、提示用户输入该接口参数类型的配置信息。

[0214] 其中,接口参数类型的配置信息包括该接口参数类型中各基本参数类型的配置信

息,基本参数类型的配置信息包括基本参数类型的各属性值。

[0215] 1006、提示用户是否共享该接口参数类型的配置信息,如果接收到用户指示共享的指令,执行 1007,否则,执行 1009。

[0216] 1007、将该接口参数类型的配置信息保存到本地的共享单元,本地的共享单元将该接口参数类型的配置信息发送到中央控制器,执行 1009。

[0217] 1008、从中央控制器中获取该接口参数类型的配置信息。

[0218] 1009、根据接口参数类型的配置信息、接口参数类型的结构和接口寻址信息等,封装接口。

[0219] 1010、当用户需要测试多个接口时,接收到用户输入的指示增加新接口的指令,在本地的共享单元中查找是否有用户指示增加的接口,如果否,执行 1011,如果是,执行 1015。

[0220] 1011、到中央控制器查找是否有用户指示增加的接口,如果没有,执行 1012,如果有,执行 1015。

[0221] 1012、配置新接口。

[0222] 1013、提示用户是否需要共享该新的接口,如果用户指示共享该接口,则执行 1014,否则,执行 1016。

[0223] 1014、将该接口放入本地的共享单元,本地的共享单元将接口发送给中央控制器保存,执行 1016。

[0224] 1015、从中央控制器中获取用户指示增加的接口。

[0225] 1016、接收用户指定的多个接口间的调用关系,按照调用关系将多个接口封装成测试逻辑结构。

[0226] 1017、接收到用户输入的指示增加新测试逻辑结构的指令,在本地的共享单元中查找是否有用户指示增加的新测试逻辑结构,如果否,执行 1018;如果是,执行 1023。

[0227] 1018、到中央控制器查找是否有用户指示增加的新测试逻辑结构,如果没有,执行 1019,否则执行 1022。

[0228] 1019、配置新的测试逻辑结构。

[0229] 1020、提示用户是否需要共享该新的测试逻辑结构,如果用户指示共享,则执行 1021,如果否执行 1023。

[0230] 1021、将该新的测试逻辑结构放入本地的共享单元,本地的共享单元将新的测试逻辑结构发送给中央控制器保存,执行 1023。

[0231] 1022、从中央控制器中获取用户指示增加的测试逻辑结构。

[0232] 1023、获取测试逻辑结构中各基本对象对应的测试数据,根据测试逻辑结构和各基本对象对应的测试数据,封装测试用例。

[0233] 1024、展现用例调度配置界面,接收用户在用例调度配置界面中指定的测试用例运行规则,根据用户指定的测试用例运行规则,运行测试用例。

[0234] 1025、根据测试用例的运行结果生成测试报告。

[0235] 该实施例使用户在集成测试环境中,不需要用户写入任何代码,而是仅需要用户根据界面的提示信息进行选择,实现集成测试场景中对多个接口进行测试;同时在用户增加/删除/修改用例时都不需要变更测试代码,方便用户操作,节省接口测试的时间开销;

进一步,采用图像化的方式供用户配置测试用例及测试数据,方便用户操作,从而可以实现任何系统的集成测试;进一步,共享单元可以共享接口的参数对象、接口、测试逻辑结构、用例等,方便多人同时进行接口测试。

[0236] 本领域普通技术人员可以理解实现上述实施例方法中的全部或部分步骤是可以通程序来指令相关的硬件完成,该程序可以存储于一种计算机可读存储介质中,例如只读存储器,磁盘或光盘等。

[0237] 以上对本发明实施例所提供的接口测试方法及装置进行了详细介绍,本文中应用了具体个例对本发明的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本发明的方法及其核心思想;同时,对于本领域的一般技术人员,依据本发明的思想,在具体实施方式及应用范围上均会有改变之处,综上所述,本说明书内容不应理解为对本发明的限制。

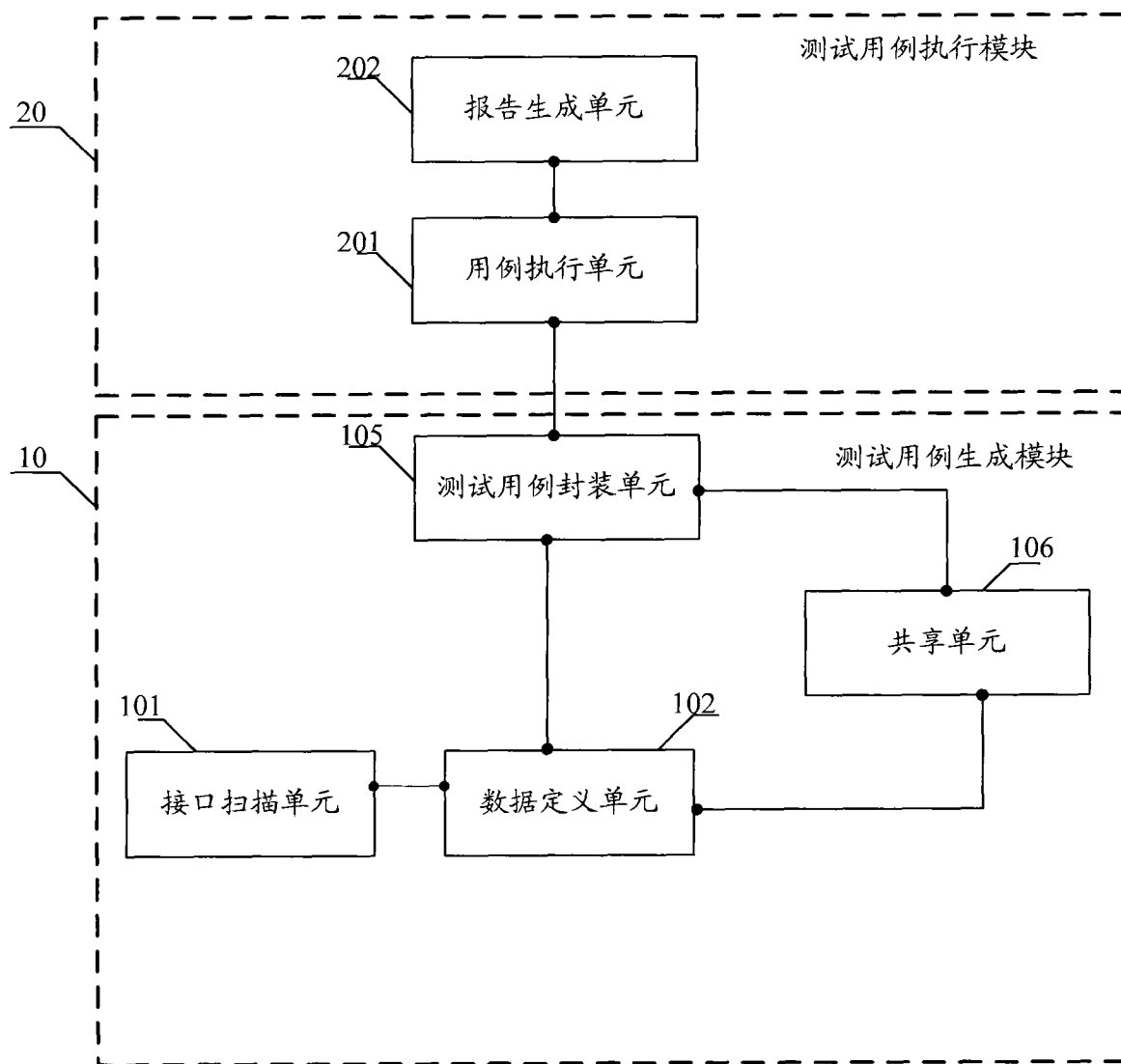


图 1

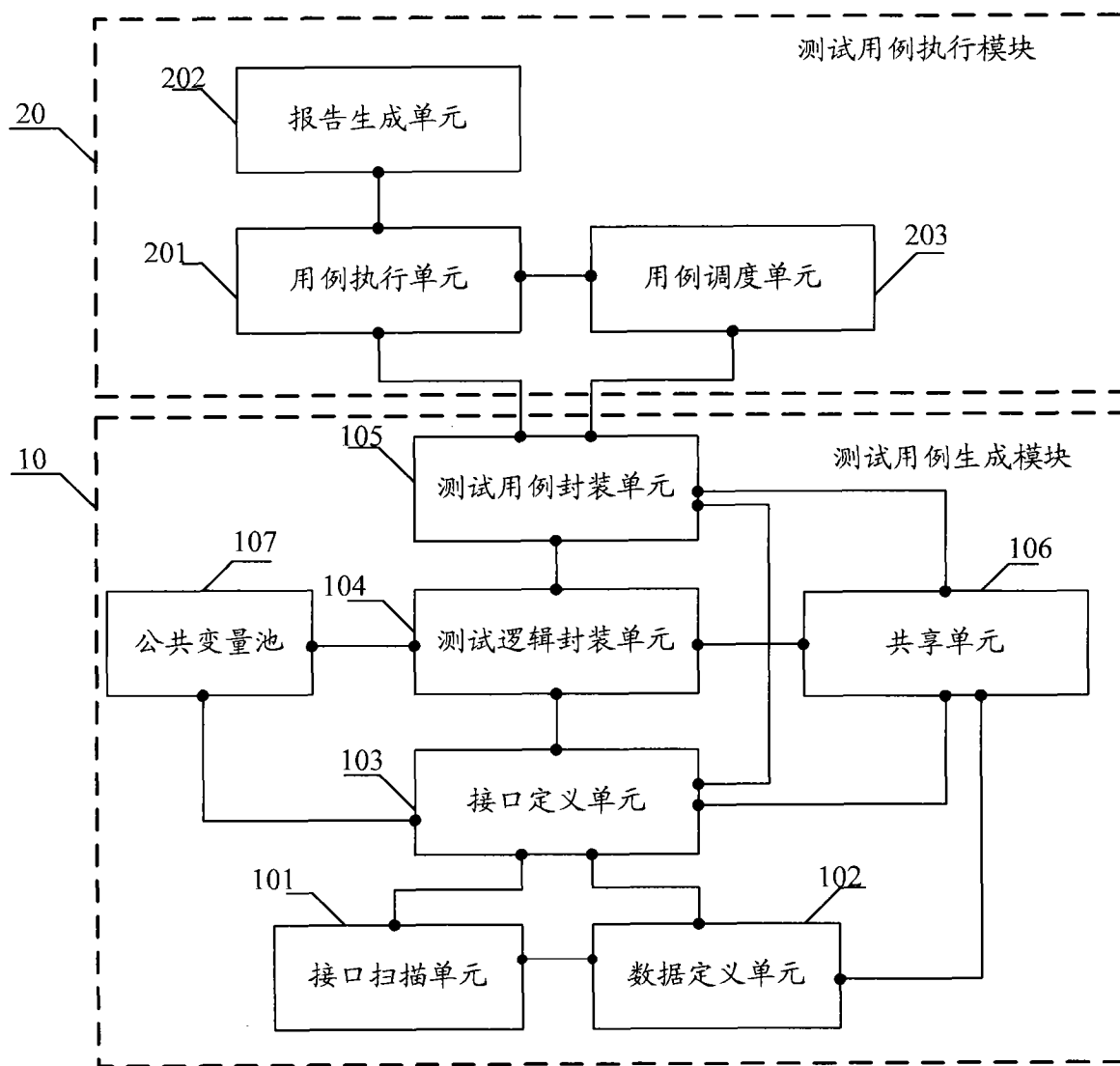


图 2A

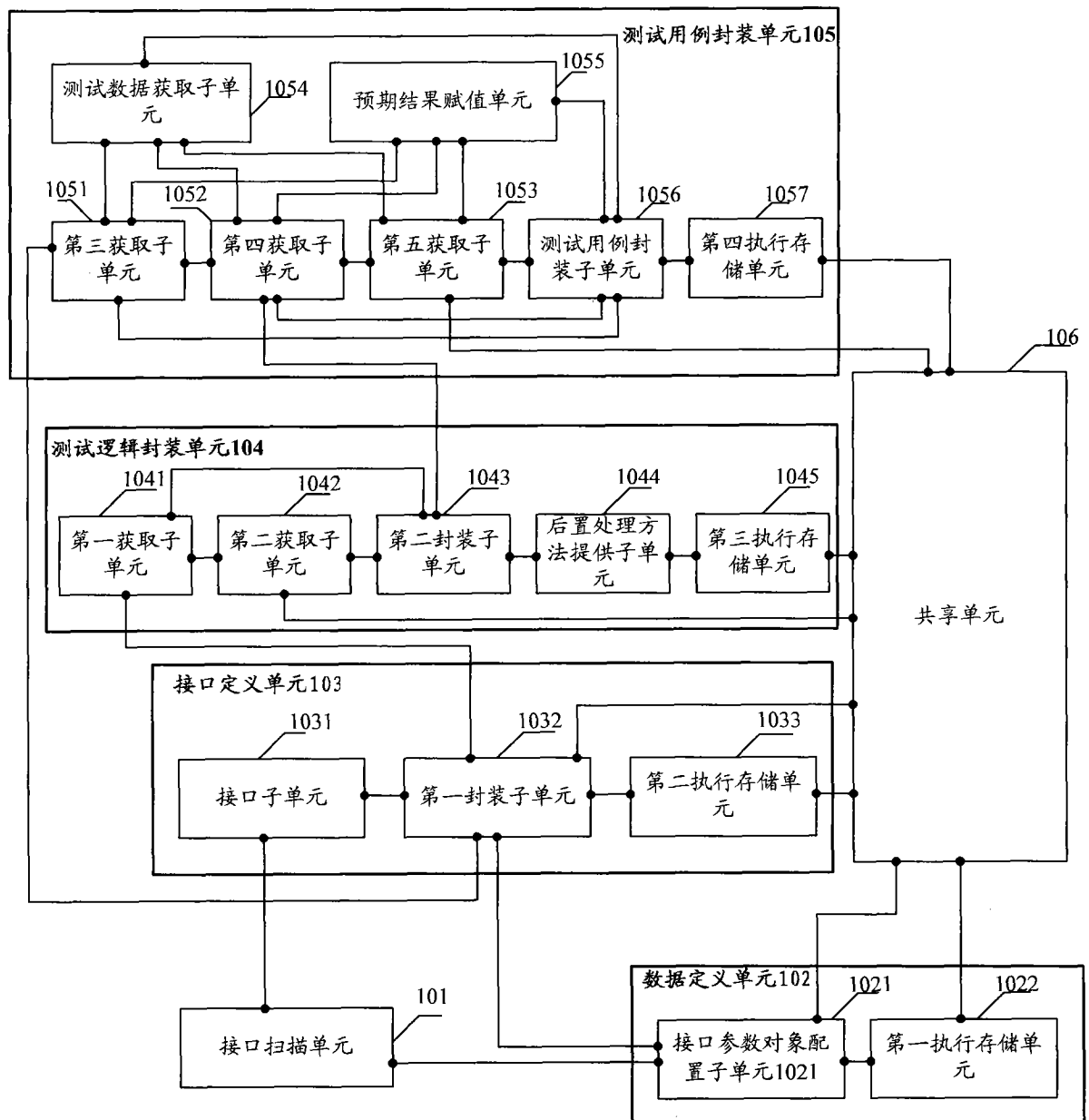


图 2B

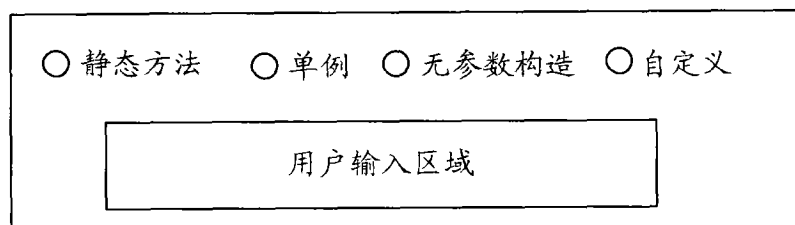


图 3A

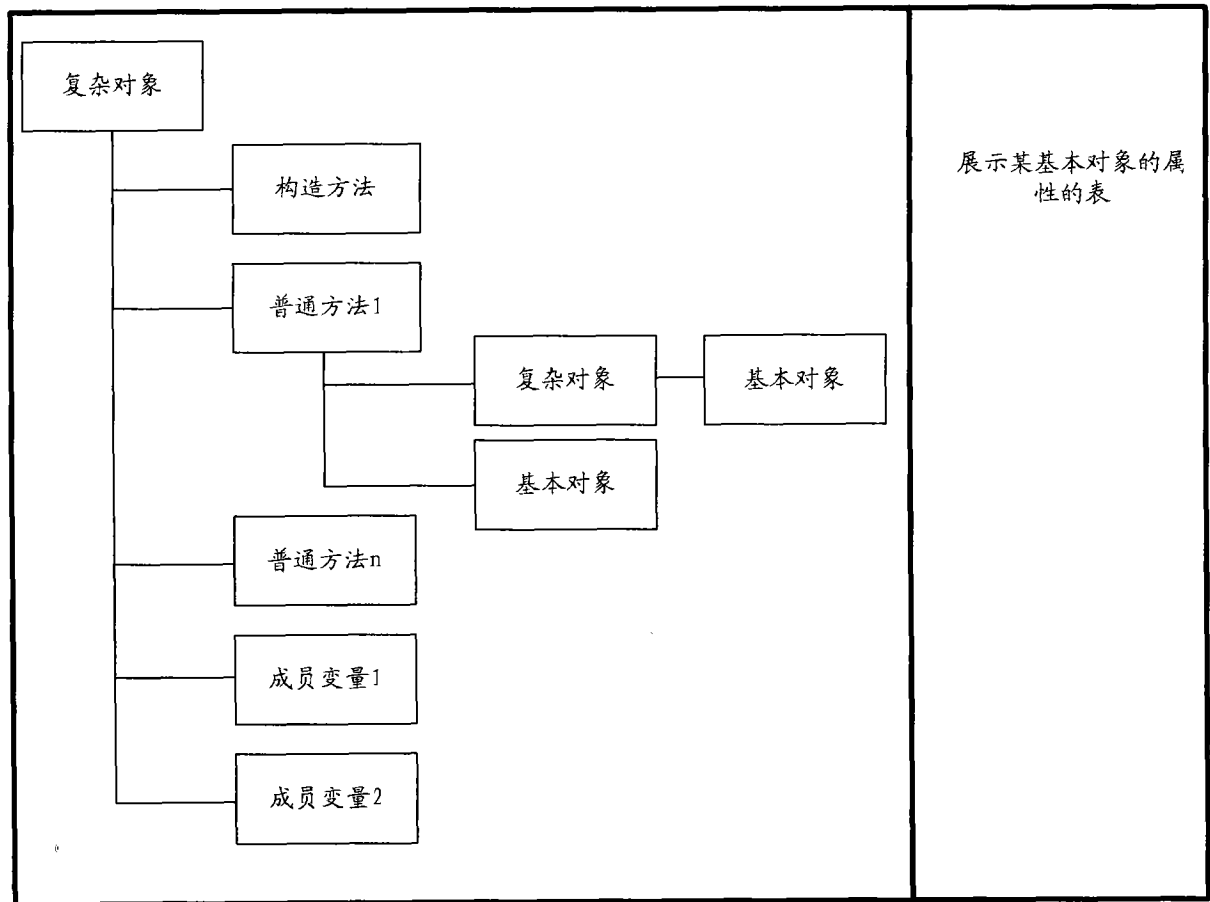


图 3B

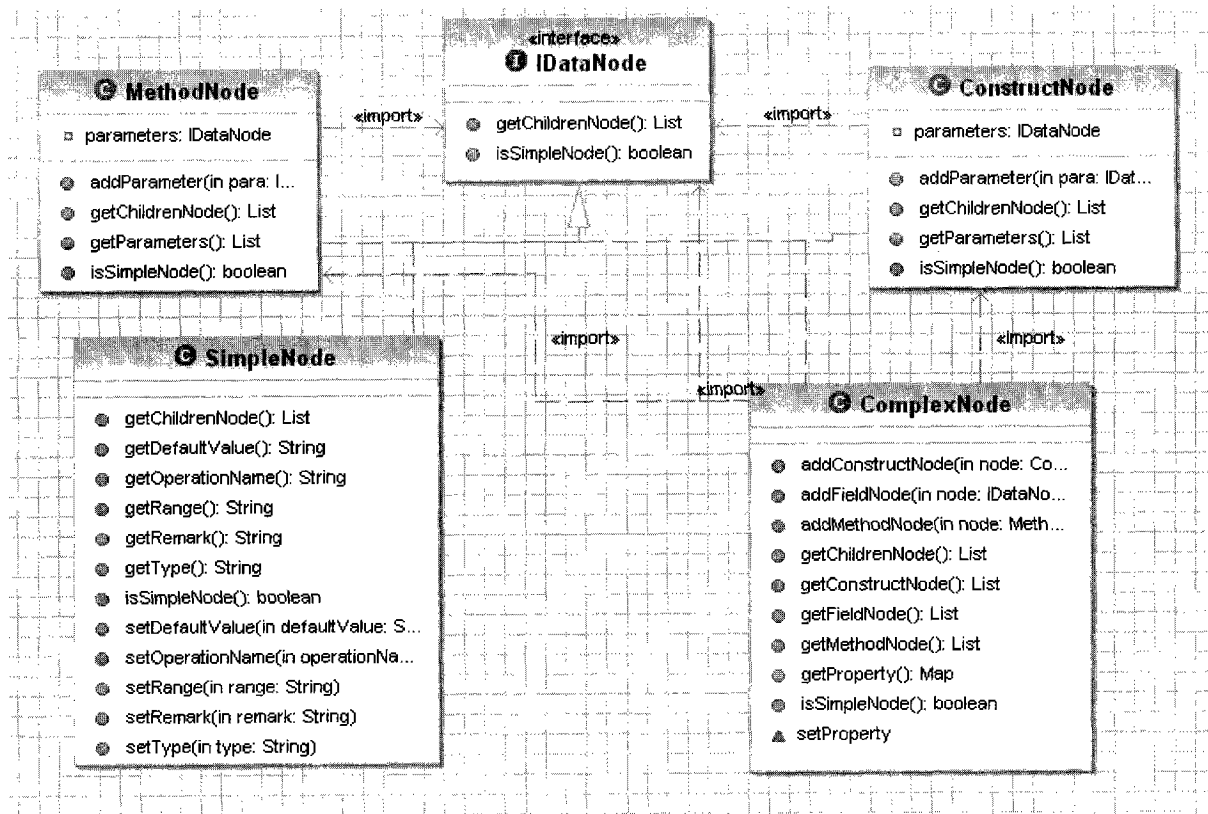


图 3C

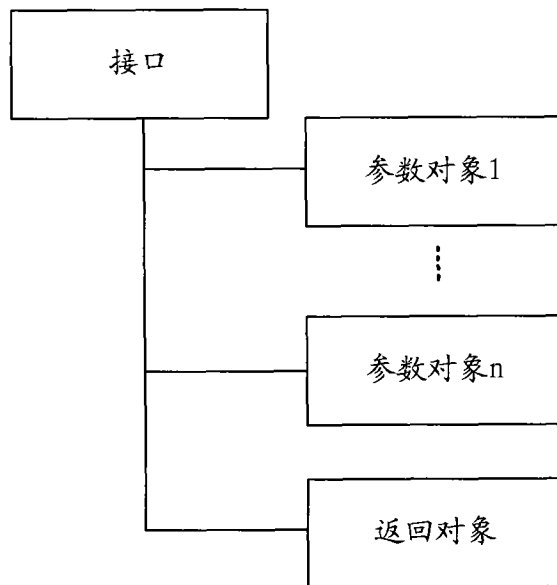


图 4A

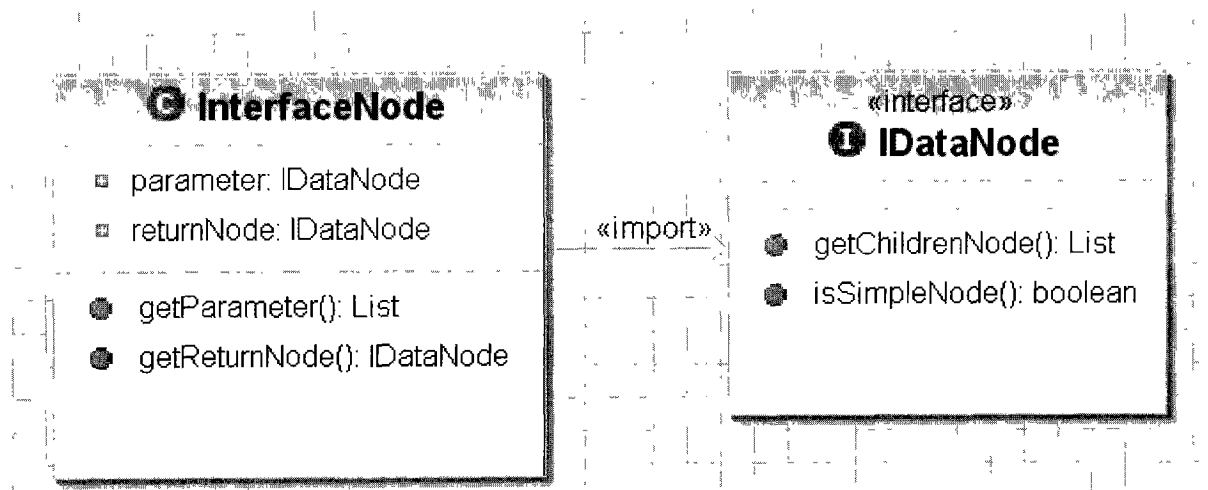


图 4B

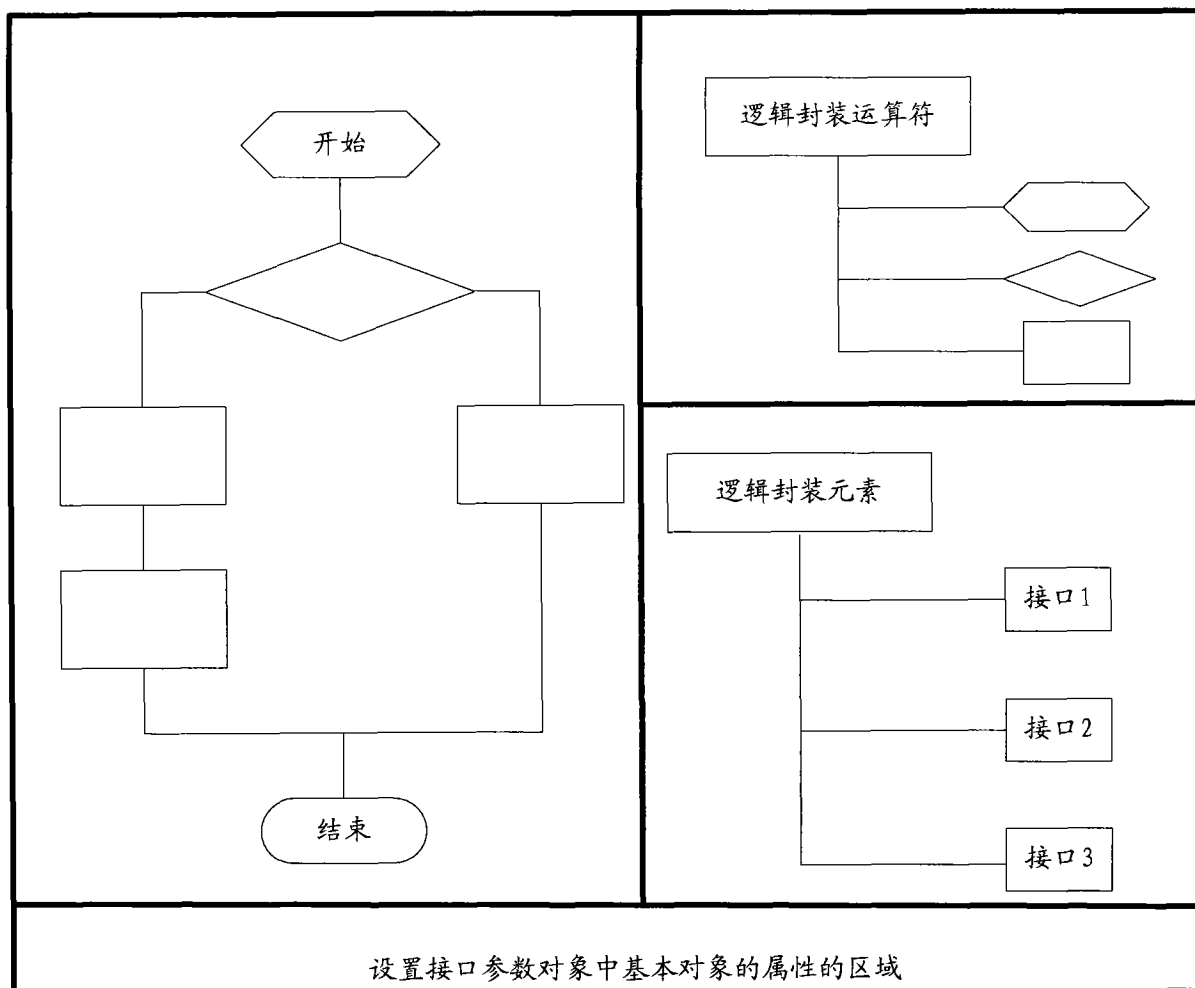


图 5A

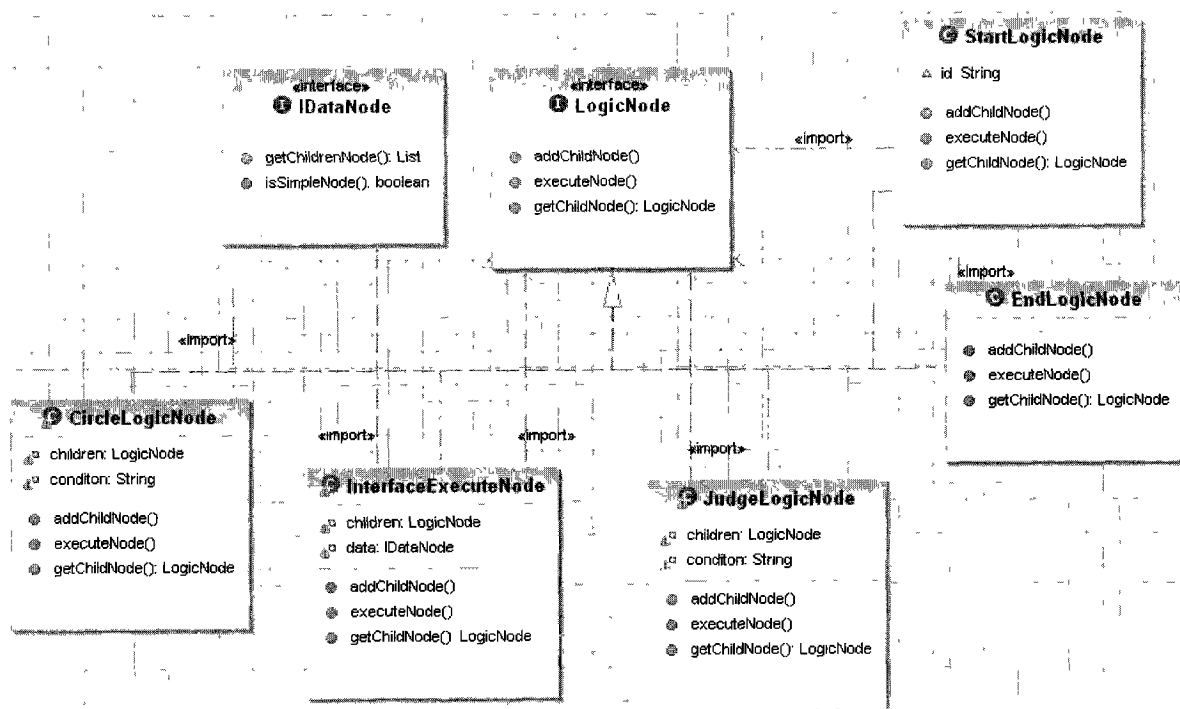


图 5B

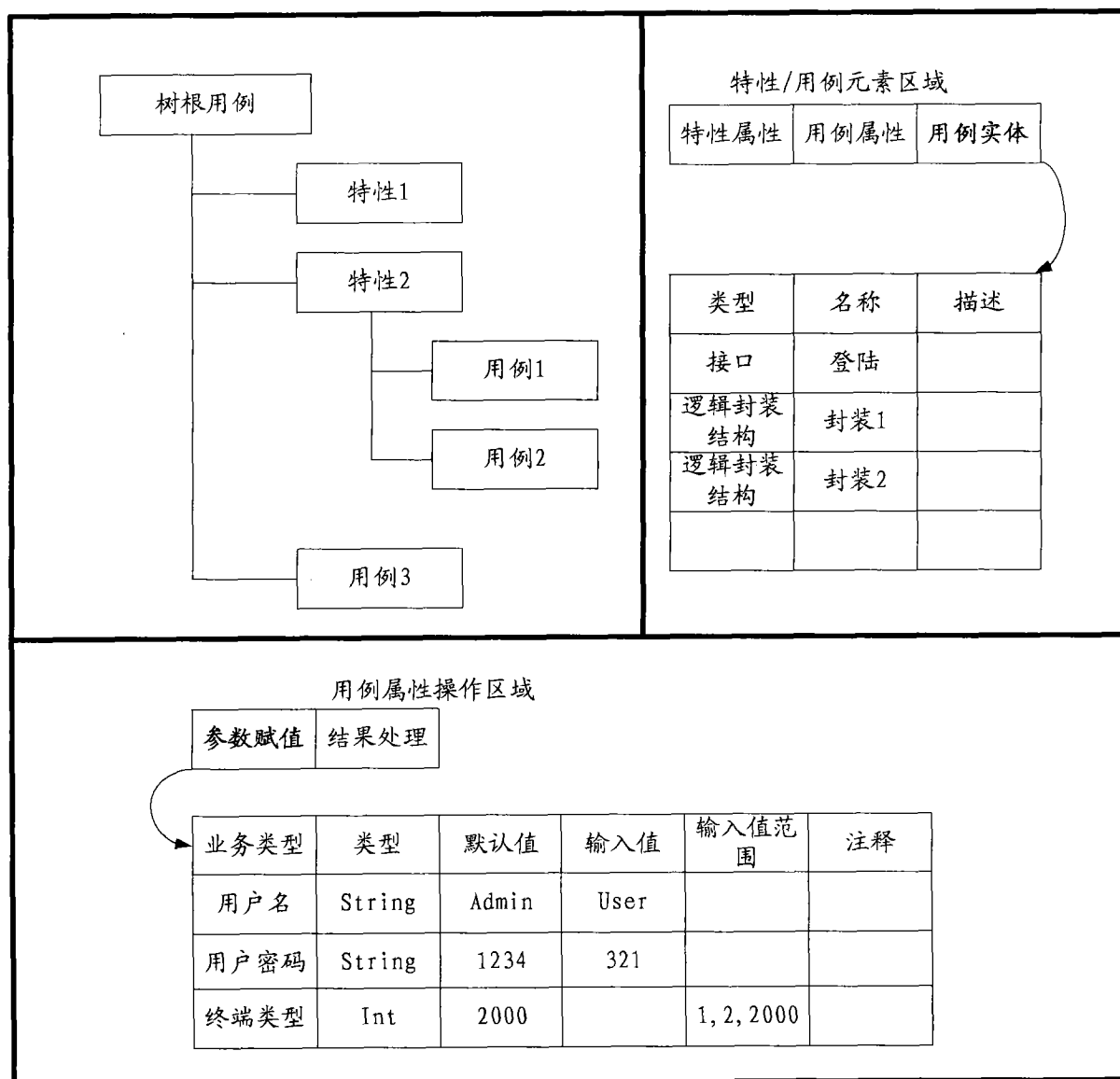


图 6A

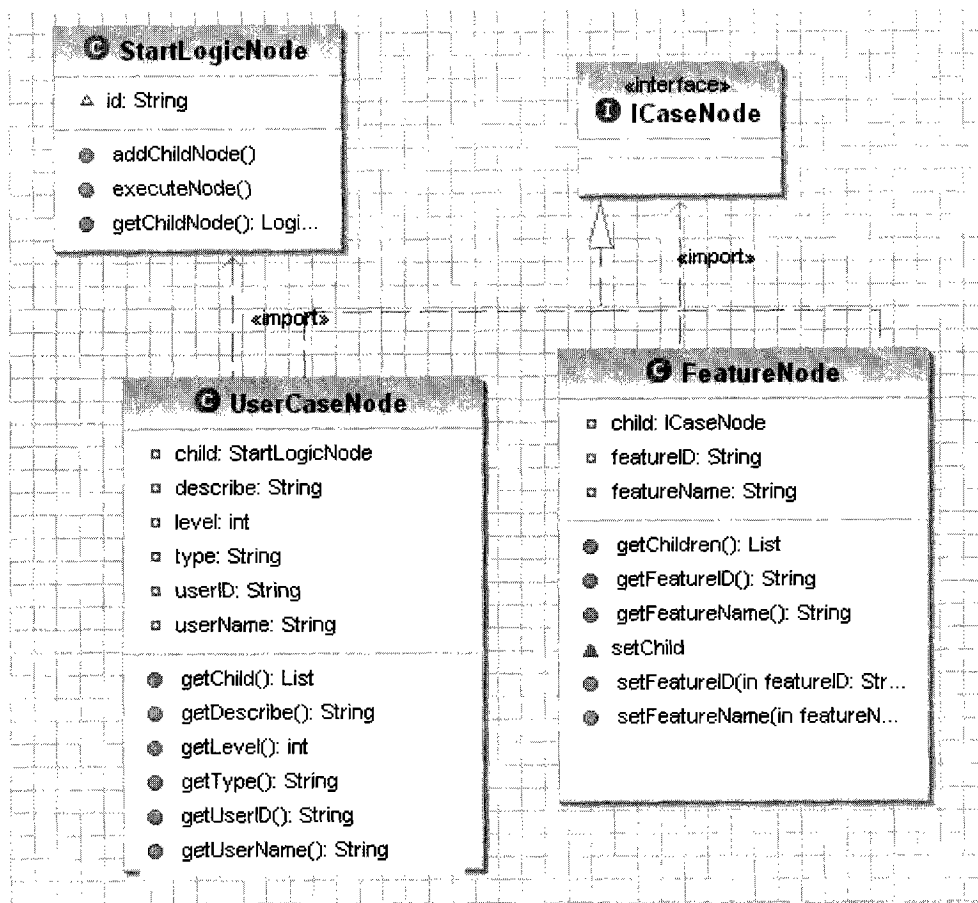


图 6B

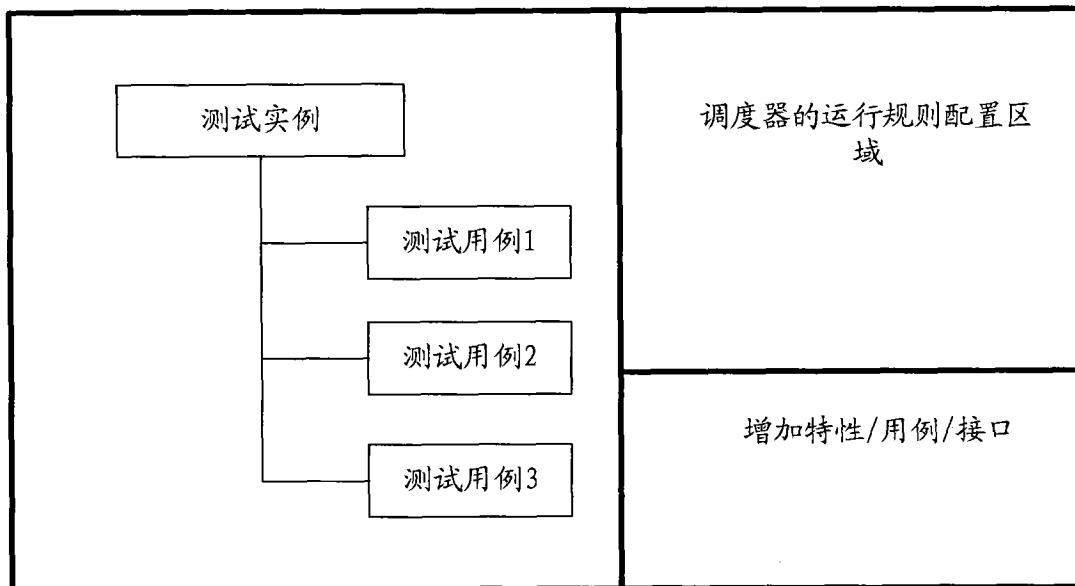


图 7A

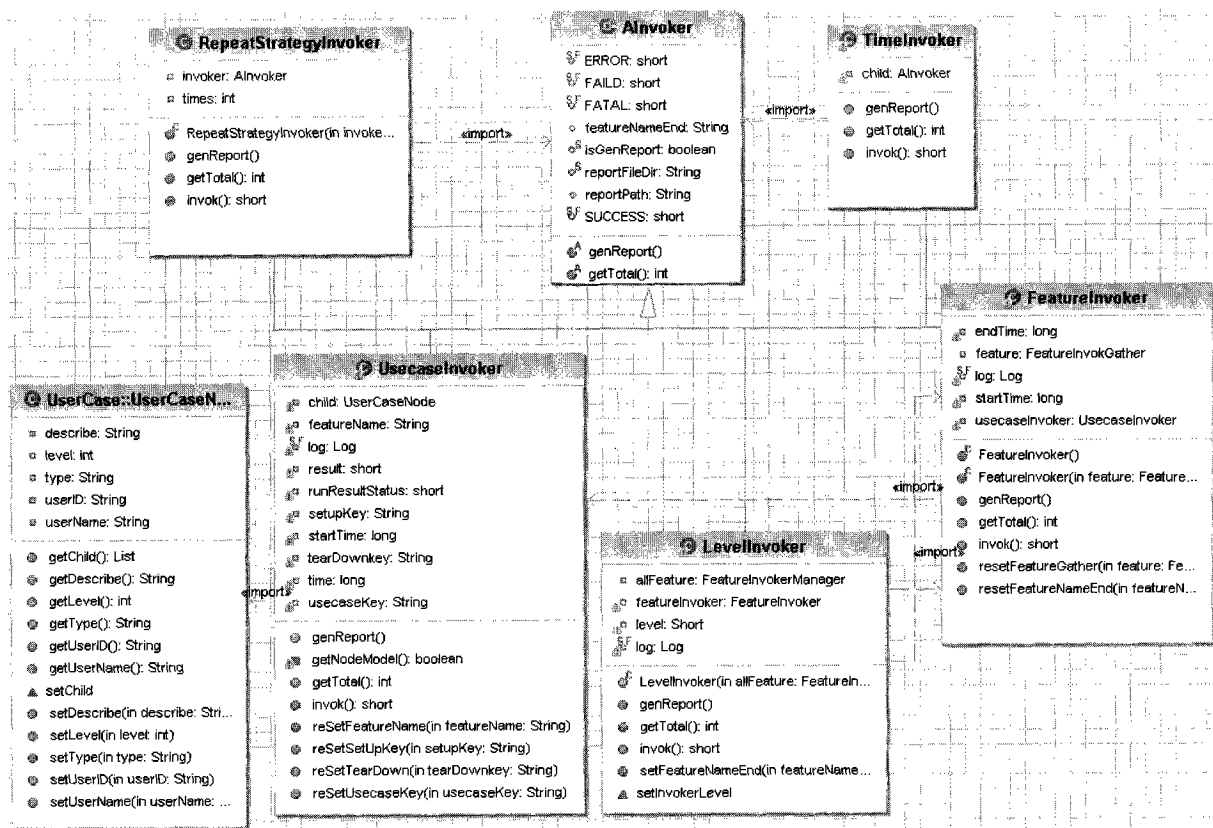


图 7B

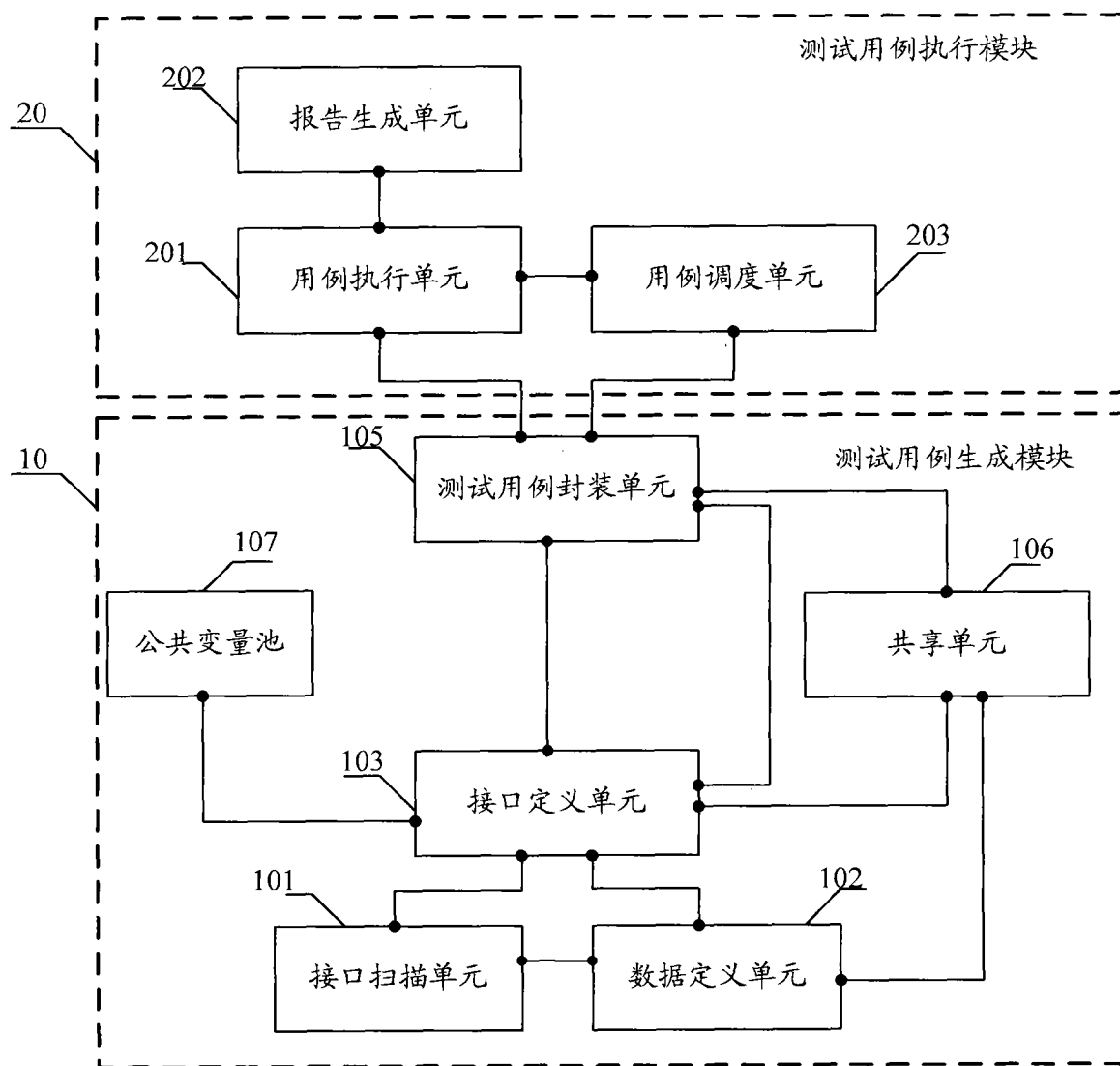


图 8

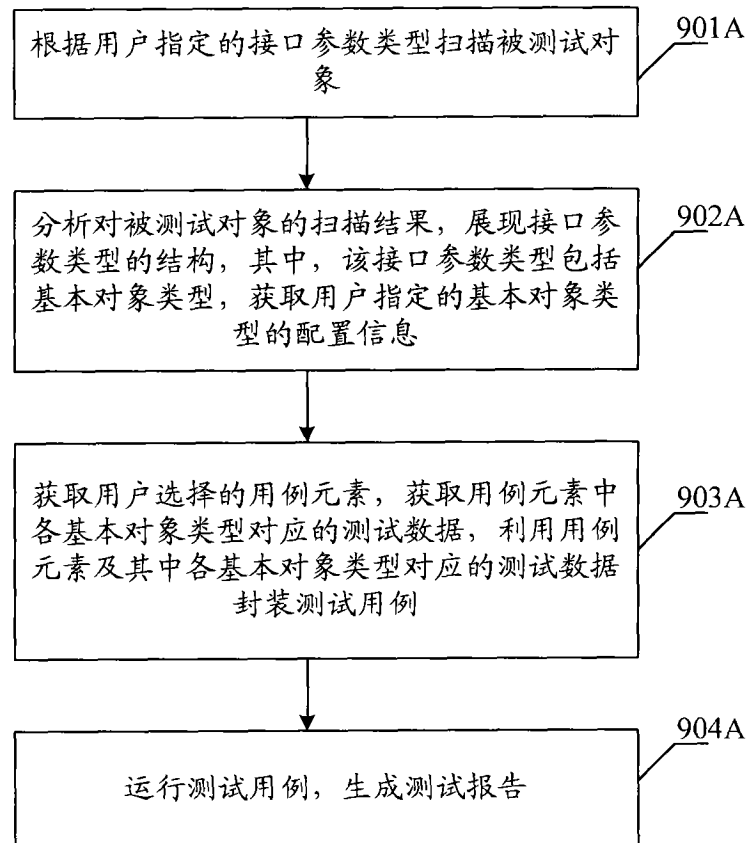


图 9A

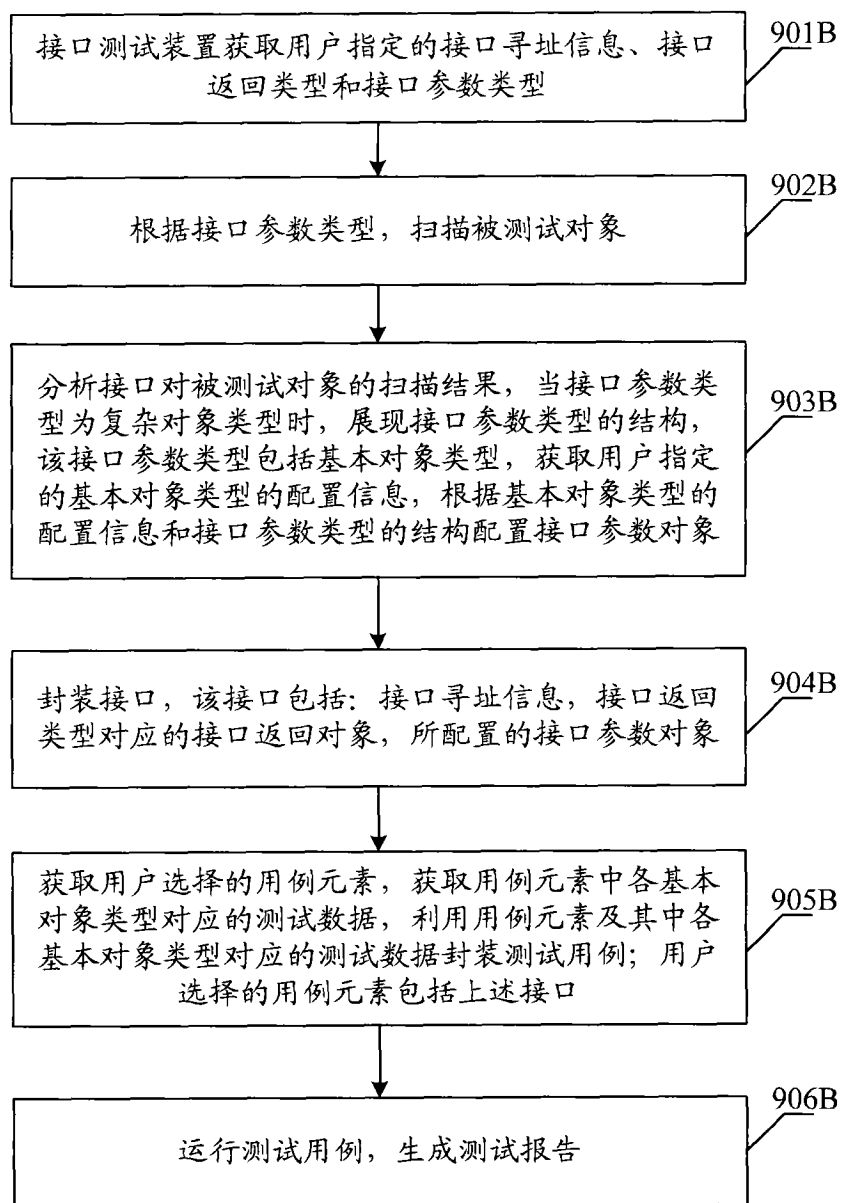


图 9B

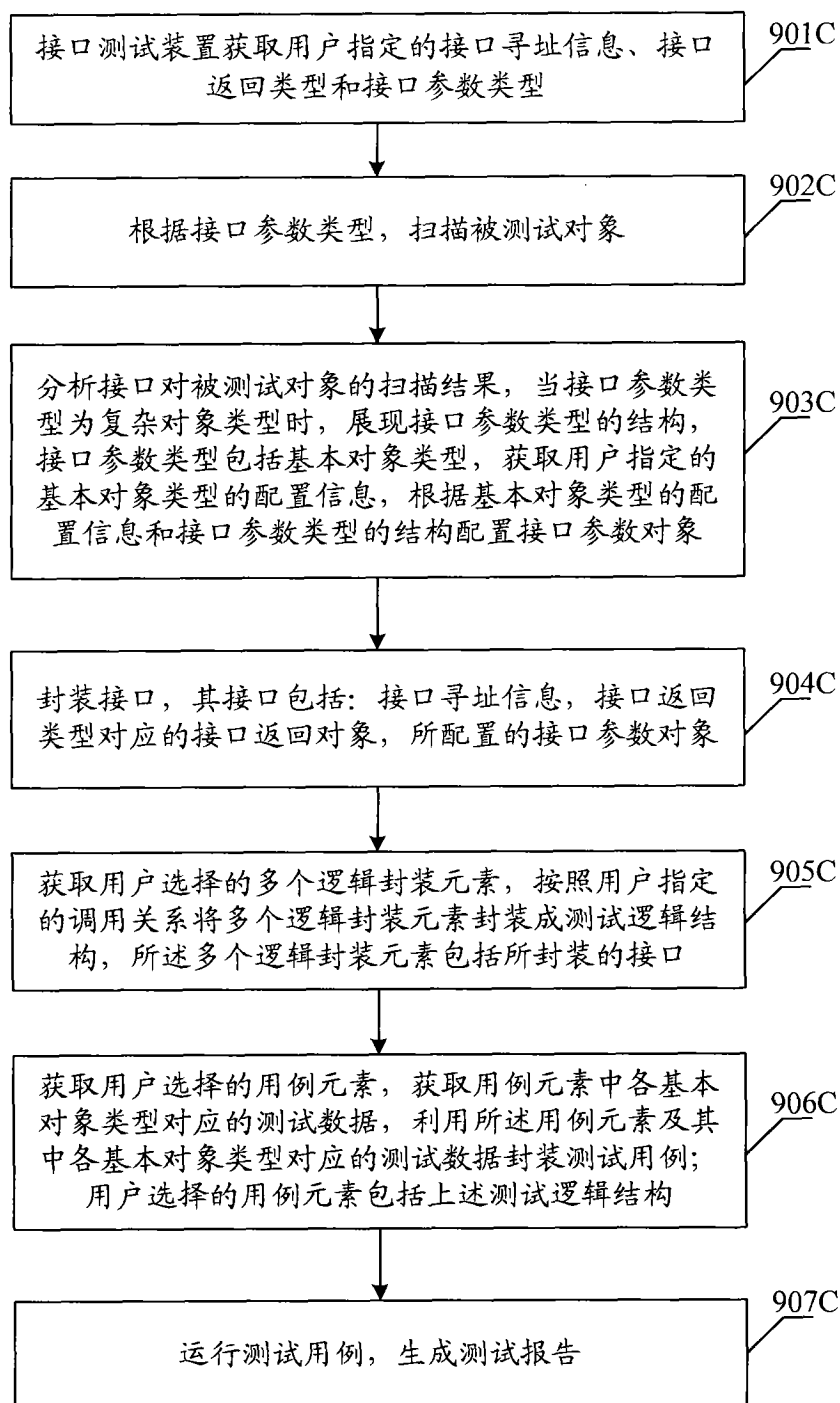


图 9C

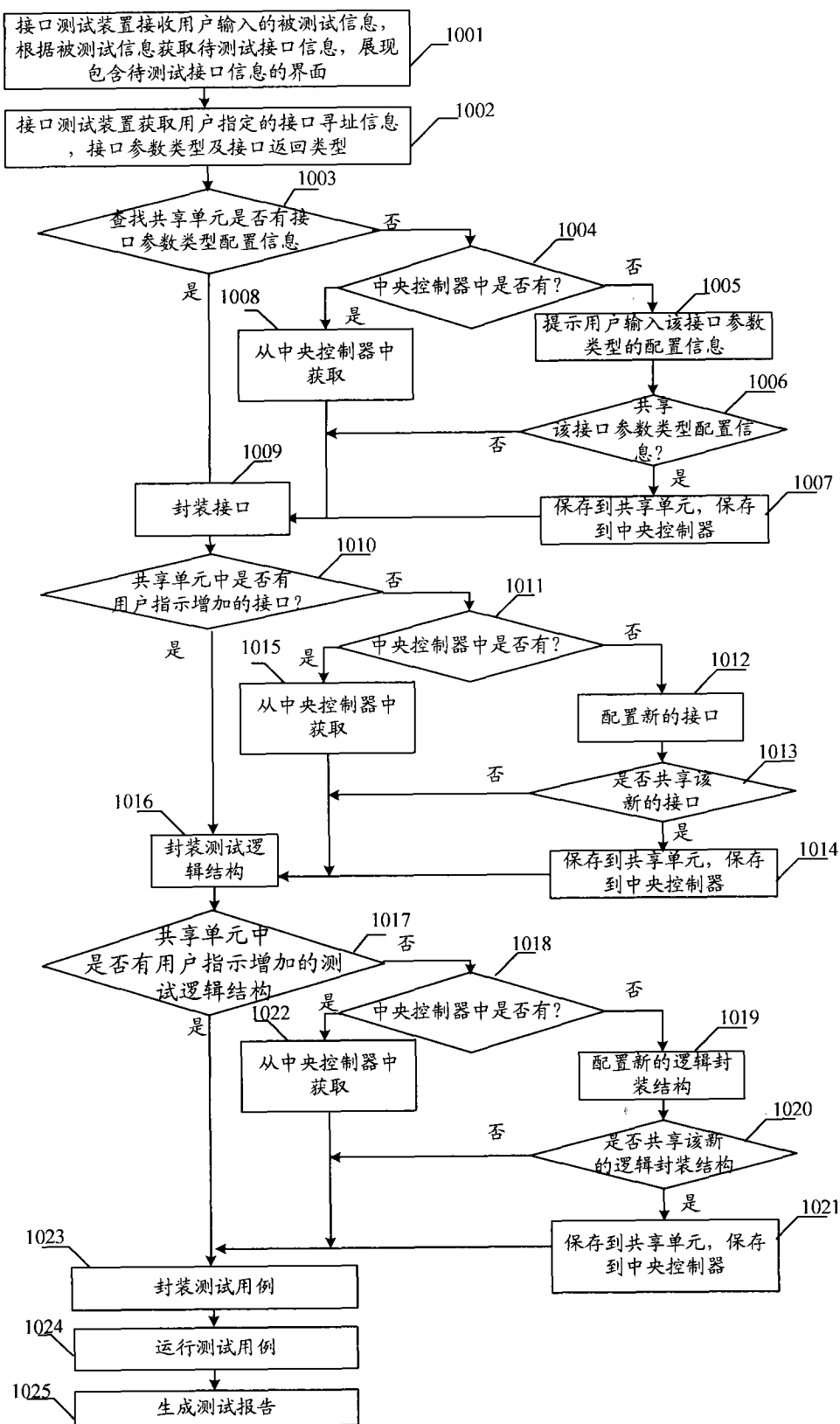


图 10