



(21) 申请号 202011608571.5

(22) 申请日 2020.12.30

(65) 同一申请的已公布的文献号  
申请公布号 CN 112597051 A

(43) 申请公布日 2021.04.02

(73) 专利权人 北京华胜天成科技股份有限公司  
地址 100093 北京市海淀区西北旺东路10  
号院东区23号楼5层501

(72) 发明人 孙文博

(74) 专利代理机构 北京林达刘知识产权代理事  
务所(普通合伙) 11277  
专利代理师 刘新宇

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

(56) 对比文件

CN 108334449 A, 2018.07.27

审查员 李婧

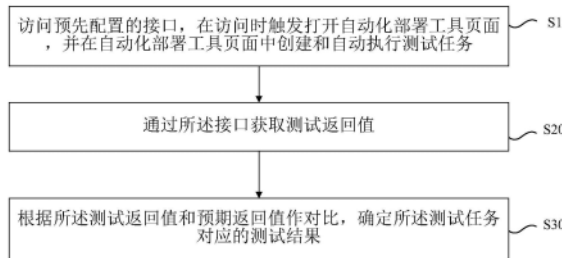
权利要求书2页 说明书9页 附图3页

#### (54) 发明名称

获取测试结果的方法及装置、电子设备和存储介质

#### (57) 摘要

本公开涉及一种获取测试结果的方法及装置、电子设备和存储介质,所述方法包括:访问预先配置的接口,在访问时触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和执行测试任务;通过所述接口获取测试返回值;根据所述测试返回值和预期返回值作对比,确定所述测试任务对应的测试结果,所述预期返回值包括所述测试任务对应的预期结果信息。本公开实施例能够获取到测试结果,尤其是能够获取到校验linux系统的底层资源创建是否成功的测试结果。



1. 一种获取测试结果的方法,其特征在于,应用于软件测试工具,包括:

访问预先配置的接口,在访问时触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和自动执行测试任务,所述测试任务用于验证linux系统的底层资源的创建情况;

通过所述接口获取测试返回值;

根据所述测试返回值和预期返回值作对比,确定所述测试任务对应的测试结果,所述测试结果包括所述测试任务对应的底层架构的构建情况,所述预期返回值包括所述测试任务对应的预期结果信息、预期的中间参数、构建的底层资源的信息;

其中,所述通过所述接口获取测试返回值,包括:通过所述接口内的UI自动化程序定位和获取所述自动化部署工具页面中的输出信息;通过所述接口获取所述输出信息作为测试返回值;

其中,所述软件测试工具为postman软件,所述接口为第三方库flask开发的供所述postman软件访问的flask接口,所述flask接口内有采用UI自动化工具开发的selenium程序,所述selenium程序中写有用于执行测试的业务逻辑,所述自动化部署工具页面为jenkins WEB页面;

其中,所述访问预先配置的接口,在访问时触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和自动执行测试任务,包括:

利用所述postman软件访问所述flask接口以触发所述selenium程序,在访问时通过所述selenium程序触发打开所述jenkins WEB页面,利用所述selenium程序在所述jenkinsWEB页面中创建所述测试任务,并根据所述selenium程序中的与测试任务相关的业务逻辑在所述linux系统中执行所述测试任务,并在执行完成后返回所述测试返回值至所述jenkins WEB页面。

2. 根据权利要求1所述的方法,其特征在于,所述接口的数量为多个;

不同所述接口内的UI自动化程序执行的业务逻辑不同,用于创建和执行不同的测试任务。

3. 根据权利要求1所述的方法,其特征在于,所述根据所述测试返回值和预期返回值作对比,确定所述测试任务对应的测试结果,还包括:

响应于所述测试返回值和预期返回值是否一致,确定测试结果为成功或失败。

4. 一种获取测试结果的装置,其特征在于,应用于软件测试工具,包括:

触发模块,用于在访问预先配置的接口时触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和自动执行测试任务,所述测试任务用于验证linux系统的底层资源的创建情况;

返回模块,用于通过所述接口获取测试返回值;

对比模块,用于根据所述测试返回值和预期返回值作对比,确定所述测试任务的测试结果,所述测试结果包括所述测试任务对应的底层架构的构建情况,所述预期返回值包括所述测试任务对应的预期结果信息、预期的中间参数、构建的底层资源的信息;

其中,所述返回模块,还用于通过所述接口内的UI自动化程序定位和获取所述自动化部署工具页面中的输出信息;再通过所述接口获取所述输出信息作为测试返回值;

其中,所述软件测试工具为postman软件,所述接口为第三方库flask开发的供所述

postman软件访问的flask接口,所述flask接口内有采用UI自动化工具开发的selenium程序,所述selenium程序中写有用于执行测试的业务逻辑,所述自动化部署工具页面为jenkins WEB页面;

其中,所述触发模块,还用于利用所述postman软件访问所述flask接口以触发所述selenium程序,在访问时通过所述selenium程序触发打开所述jenkins WEB页面,利用selenium程序在所述jenkinsWEB页面中创建所述测试任务,并根据所述selenium程序中的与测试任务相关的业务逻辑在所述linux系统中执行所述测试任务,并在执行完成后返回所述测试返回值至所述jenkins WEB页面。

5. 一种电子设备,其特征在于,包括:

处理器;

用于存储处理器可执行指令的存储器;

其中,所述处理器被配置为调用所述存储器存储的指令,以执行权利要求1-3中任意一项所述的方法。

6. 一种计算机可读存储介质,其上存储有计算机程序指令,其特征在于,所述计算机程序指令被处理器执行时实现权利要求1-3中任意一项所述的方法。

## 获取测试结果的方法及装置、电子设备和存储介质

### 技术领域

[0001] 本公开涉及软件测试技术领域,尤其涉及一种获取测试结果的方法及装置、电子设备和存储介质。

### 背景技术

[0002] 软件自动化测试是相对手工测试而存在的,由测试人员根据测试用例中描述的规程一步步执行测试,得到实际结果与期望结果的比较。采用自动化软件测试方法相对手工测试而言,节省人力、时间或硬件资源、测试效率高。现有技术中,在测试linux系统底层的资源时,利用现有的自动化测试工具无法获取测试结果,导致无法校验linux系统的底层(k8s、Ceph、apache、mysql、等.)资源创建是否成功。

### 发明内容

[0003] 本公开提供了一种获取测试结果的方法及装置、电子设备和存储介质,能够获取到测试结果,尤其是能够获取到校验linux系统的底层资源创建是否成功的测试结果。

[0004] 根据本公开的第一方面,提供了一种获取测试结果的方法,其包括:

[0005] 访问预先配置的接口,在访问时触发打开自动化部署工具页面,在所述自动化部署工具页面中创建和自动执行测试任务;

[0006] 通过所述接口获取测试返回值;

[0007] 根据所述测试返回值和预期返回值作对比,确定所述测试任务对应的测试结果,所述预期返回值包括所述测试任务对应的预期结果信息。

[0008] 在一些可能的实施方式中,所述接口内采用UI自动化工具开发,在访问时通过所述接口内的UI自动化程序触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和执行测试任务;所述UI自动化程序中写有用于执行测试的业务逻辑。

[0009] 在一些可能的实施方式中,所述通过所述接口获取测试返回值,包括:

[0010] 通过所述接口内的UI自动化程序定位和获取所述自动化部署工具页面中的输出信息;

[0011] 通过所述接口获取所述输出信息作为测试返回值。

[0012] 在一些可能的实施方式中,所述接口的数量为多个;

[0013] 不同所述接口内的UI自动化程序执行的业务逻辑不同,用于创建和执行不同的测试任务。

[0014] 在一些可能的实施方式中,所述根据所述测试返回值和预期返回值作对比,确定所述测试任务对应的测试结果,还包括:

[0015] 响应于所述测试返回值和预期返回值是否一致,确定测试结果为成功或失败。

[0016] 根据本公开的第二方面,提供了一种获取测试结果的装置,其包括:

[0017] 触发模块,用于在访问预先配置的接口时触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和自动执行测试任务;

- [0018] 返回模块,用于通过所述接口获取测试返回值;
- [0019] 对比模块,用于根据所述测试返回值和预期返回值作对比,确定所述测试任务的测试结果,所述预期返回值包括所述测试任务对应的预期结果信息。
- [0020] 在一些可能的实施方式中,所述接口内采用UI自动化工具开发;
- [0021] 所述触发模块,还用于在访问时通过所述接口内的UI自动化程序触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和自动执行测试任务;;所述UI自动化程序中写有用于执行测试的业务逻辑。
- [0022] 在一些可能的实施方式中,所述返回模块,还用于通过所述接口内的UI自动化程序定位和获取所述自动化部署工具页面中的输出信息;再通过所述接口获取所述输出信息作为测试返回值。
- [0023] 根据本公开的第三方面,提供了一种电子设备,其包括:
- [0024] 处理器;
- [0025] 用于存储处理器可执行指令的存储器;
- [0026] 其中,所述处理器被配置为调用所述存储器存储的指令,以执行第一方面中任意一项所述的方法。
- [0027] 根据本公开的第四方面,提供了一种计算机可读存储介质,其上存储有计算机程序指令,所述计算机程序指令被处理器执行时实现第一方面中任意一项所述的方法。
- [0028] 在本公开实施例中,可以利用程序测试工具访问创建的接口,通过访问接口实现在自动化部署工具页面中创建和执行测试任务,并通过该接口获得测试返回值;再利用测试返回值和预期返回值作对比,确定所述测试任务对应的测试结果。本公开实施例能够获取到测试结果,尤其是能够获取到校验linux系统的底层资源创建是否成功的测试结果。
- [0029] 应当理解的是,以上的一般描述和后文的细节描述仅是示例性和解释性的,而非限制本公开。
- [0030] 根据下面参考附图对示例性实施例的详细说明,本公开的其它特征及方面将变得清楚。

### 附图说明

- [0031] 此处的附图被并入说明书中并构成本说明书的一部分,这些附图示出了符合本公开的实施例,并与说明书一起用于说明本公开的技术方案。
- [0032] 图1示出根据本公开实施例的一种获取测试结果的方法的流程图;
- [0033] 图2示出根据本公开实施例的获取测试结果的方法中步骤S20的流程图;
- [0034] 图3示出根据本公开实施例的获取测试结果装置的框图;
- [0035] 图4示出根据本公开实施例的一种电子设备的框图;
- [0036] 图5示出根据本公开实施例的另一种电子设备的框图。

### 具体实施方式

- [0037] 以下将参考附图详细说明本公开的各种示例性实施例、特征和方面。附图中相同的附图标记表示功能相同或相似的元件。尽管在附图中示出了实施例的各种方面,但是除非特别指出,不必按比例绘制附图。

[0038] 在这里专用的词“示例性”意为“用作例子、实施例或说明性”。这里作为“示例性”所说明的任何实施例不必解释为优于或好于其它实施例。

[0039] 本文中术语“和/或”，仅仅是一种描述关联对象的关联关系，表示可以存在三种关系，例如，A和/或B，可以表示：单独存在A，同时存在A和B，单独存在B这三种情况。另外，本文中术语“至少一种”表示多种中的任意一种或多种中的至少两种的任意组合，例如，包括A、B、C中的至少一种，可以表示包括从A、B和C构成的集合中选择的任意一个或多个元素。

[0040] 另外，为了更好地说明本公开，在下文的具体实施方式中给出了众多的具体细节。本领域技术人员应当理解，没有某些具体细节，本公开同样可以实施。在一些实例中，对于本领域技术人员熟知的方法、手段、元件和电路未作详细描述，以便于凸显本公开的主旨。

[0041] 获取测试结果的方法的执行主体可以是任意可以用于测试的装置，例如，方法可以由终端设备或服务器或其它处理设备执行，其中，终端设备可以为用户设备 (User Equipment, UE)、移动设备、用户终端、终端、蜂窝电话、无绳电话、个人数字处理 (Personal Digital Assistant, PDA)、手持设备、计算设备、车载设备、可穿戴设备等。在一些可能的实现方式中，该方法可以通过处理器调用存储器中存储的计算机可读指令的方式来实现。

[0042] 可以理解，本公开提及的上述各个方法实施例，在不违背原理逻辑的情况下，均可以彼此相互结合形成结合后的实施例，限于篇幅，本公开不再赘述。

[0043] 图1示出根据本公开实施例的获取测试结果的方法的流程图，如图1所示，所述获取测试结果方法包括：

[0044] S10:访问预先配置的接口，在访问时触发打开自动化部署工具页面，并在所述自动化部署工具页面中创建和执行测试任务；

[0045] S20:通过所述接口获取测试返回值；

[0046] S30:根据所述测试返回值和预期返回值作对比，确定所述测试任务对应的测试结果，所述预期返回值包括所述测试任务对应的预期结果信息。

[0047] 本公开实施例中，获取测试结果方法的执行主体为程序测试工具，程序测试工具优选为接口测试工具Postman，相关技术中，为了测试linux系统的底层资源是否创建成功，采用的postman软件进行自动化测试，在测试过程中发现：无法获取测试结果，因而导致无法校验linux系统的底层 (k8s、Ceph、apache、mysql等) 资源创建是否成功。因此，本公开实施例结合自动化部署工具页面和预先配置的接口，配合获得测试返回值，用以验证linux系统的底层资源是否创建成功。当然，在实际应用中，软件测试工具还可以采用其他相应的工具，比如：jmeter等。

[0048] 本公开实施例在执行测试之前，即利用程序测试工具访问预先配置的接口之前，包括：用户配置该接口，该接口优选为：flask接口，即。当然，在实际应用中还可以采用其他类型的接口，例如：django接口。

[0049] 本公开实施例的接口内可以采用UI自动化工具开发 (即，该接口内有IU自动化程序)，在访问时通过接口内的UI自动化程序触发打开自动化部署工具页面，并在所述自动化部署工具页面中创建和自动执行测试任务，该UI自动化程序中写有用于执行测试的业务逻辑。本实施例中，UI自动化工具优选为selenium；自动化部署工具优选为jenkins；自动化部署工具页面为jenkins WEB页面。当然，在实际应用中，本发明并不局限于此，UI自动化工具还可以为Robot Framework、UTF，自动化部署工具还可以包括tekton、drone、jenkins x。

[0050] 因此,本公开实施例使用第三方库flask开发供程序测试工具postman软件访问的flask接口,在flask接口中可通过selenium程序操作自动化部署工具页面jenkins WEB页面,目的是为了程序测试工具postman访问flask接口时,触发接口中的selenium程序,执行打开自动化部署工具、创建测试任务、执行测试任务等操作。

[0051] 其中,Selenium是用于Web应用程序测试的工具。Selenium测试直接运行在浏览器中,就像真正的用户在操作一样。Flask是使用Python语言编写的轻量级Web应用框架; Jenkins是一个开源软件项目,是基于Java开发的一种持续集成工具,用于监控持续重复的工作,旨在提供一个开放易用的软件平台,使软件的持续集成变成可能。

[0052] 通过上述配置,可以通过第三方库flask创建供程序测试工具postman访问的接口,在该接口内利用UI自动化工具selenium开发。本公开实施例程序测试工具postman访问接口,用于打开自动化部署工具页面,定位自动化部署工具页面执行的任务节点以及获取自动化部署工具页面内生成的测试结果等信息。通过postman访问该接口,可以调用自动化部署工具页面自动执行测试任务和获取测试结果。具体地,postman访问flask接口触发selenium程序,通过selenium程序打开jenkins WEB页面,selenium程序驱动在jenkins WEB页面的job中创建一个新job任务(测试任务)。而后可以根据selenium程序中的与测试任务相关的业务逻辑信息,执行测试任务,使该job任务在linux系统进行执行,执行完成后返回执行结果给jenkins WEB页面。

[0053] 在执行软件程序测试的过程中,可以通过程序测试工具postman访问创建的接口。本公开实施例中,创建的接口数量可以为多个,不同接口可以用于完成应用程序的不同测试,例如可以包括用于测试linux系统底层下的k8s资源是否创建成功、用于测试linux系统底层下的mysql资源是否创建成功等,本公开对此不做具体限定。而且本公开实施例中的不同接口是指不同接口内的UI自动化程序执行的业务逻辑不同,可以用于创建和执行不同的测试任务。

[0054] 本公开实施例中,程序测试工具postman在访问接口打开自动化部署工具页面时,还包括:需要获取验证信息,该验证信息可以包括用户名和密码,只要在输入的验证信息与预先注册的信息匹配的情况下,自动化部署工具页面才被打开。

[0055] 具体的,图2示出根据本公开实施例的获取测试结果的方法中步骤S20的流程图。其中,所述通过所述接口获取测试返回值,包括:

[0056] S21:通过所述接口内的UI自动化程序定位和获取所述自动化部署工具页面中的输出信息;

[0057] S22:通过所述接口获取所述输出信息作为测试值返回。

[0058] 具体地,可以通过selenium定位和获取jenkins WEB页面中的控制台输出的内容,并通过flask接口返回给postman,使postman能够获取到返回值;本公开实施例可以实时存储上述获得的信息。本公开实施例中,job是jenkins的一个任务或者项目,可配置与可执行。在linux系统中执行任务;控制台是jenkins WEB页面中的控制台,用来打印输出日志以及输出结果。

[0059] 进一步地,在得到测试返回值的情况下,可以将测试返回值和测试任务对应的预期返回值进行比较,并根据所述测试返回值和预期返回值的对比结果,确定所述测试任务的测试结果,所述测试结果包括测试任务对应的底层架构的构建情况。

[0060] 具体的,预期返回值可以包括针对测试任务的预期的生成结果,即在测试任务在对应的应用程序上被执行时预测应当生成的结果。例如可以包括预期的中间参数以及最终的执行结果(成功或者失败)等信息,或者也可以包括构建的底层资源的信息,本公开对此不做具体限定,只要是执行测试任务能够获得的参数或者结果都可以作为预期返回值。

[0061] 在得到测试返回值的情况下,可以将测试返回值和预期返回值做比较,如果二者相同,则表示测试任务当前测试的应用程序的状态与预期一致,即测试结果为成功。如果二者存在差异,可以确定为当前测试的应用程序的状态存在故障或者错误,即测试结果为失败,此时需要改进和修改应用程序。

[0062] 在一些可能的实施方式中,预期返回值中可以包括至少一个返回值,返回值可以与应用程序的底层资源相匹配,在执行测试任务的过程中,如果相应的底层资源运行无误且被成功创建,则可以生成与该底层资源对应的返回值,即为测试返回值。因此,测试返回值可以表示与测试任务对应的应用程序中的底层资源是否被成功创建。在本公开实施例中,底层资源可以包括k8s、Ceph、apache、mysql中的至少一种。在对应的底层资源被创建时,生成的返回值可以为第一标识,在对应的底层资源创建失败时,生成的返回值可以为第二标识,或者不生成任何标识。对应的生成的测试返回值可以为由第一标识和第二标识组成的序列。其中第一标识可以为1,第二标识可以为0,本公开实施例对此不做具体限定。

[0063] 在一些可能的实时方式中,在所述预期返回值包括所述测试返回值以外的参数值的情况下,可以利用所述参数值确定创建失败的底层资源。如上所述,在底层资源被创建失败时,可以不生成与该底层资源对应的第二标识,因此,测试返回值中包括的只有被成功创建的底层资源对应的第一标识,对应的未生成的第一标识对应的底层资源则创建失败。

[0064] 在一些可能的实施方式中,在所述预期返回值与测试返回值中对应的参数值不同的情况下,可以利用不同的参数值确定创建失败的底层资源。如上所述,在底层资源创建失败时,生成与该底层资源对应的第二标识。因此,预期返回值中的各第一标识能够表示被测试的应用程序内的各底层资源被创建的预期结果,此时测试返回值和预期返回值中返回值的数量相同,创建失败的底层资源对应的第二标识与预期返回值中的第一标识不同,利用不同的标识可以确定创建失败的底层资源。相反的,如果所述测试返回值和预期返回值一致,确定所述底层资源构建成功。

[0065] 综上所述,在本公开实施例中,可以利用程序测试工具访问创建的接口,在访问时触发在自动化部署工具页面中创建和自动执行测试任务,获得测试返回值,程序测试工具利用测试返回值和预期返回值的对比结果确定测试任务的测试结果。本公开实施例能够获得程序的测试结果,尤其是能够获取到校验linux系统的底层资源创建是否成功的测试结果。

[0066] 本领域技术人员可以理解,在具体实施方式的上述方法中,各步骤的撰写顺序并不意味着严格的执行顺序而对实施过程构成任何限定,各步骤的具体执行顺序应当以其功能和可能的内在逻辑确定。

[0067] 图3示出根据本公开实施例的获取测试结果装置的框图,如图3所示,所述获取测试结果装置包括:

[0068] 触发模块10,用于在访问预先配置的接口时触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和自动执行测试任务;



[0069] 返回模块20,用于通过所述接口获取测试返回值;

[0070] 对比模块30,用于根据所述测试返回值和预期返回值作对比,确定所述测试任务的测试结果,所述预期返回值包括所述测试任务对应的预期结果信息。

[0071] 在一些可能的实施方式中,所述接口内采用UI自动化工具开发;

[0072] 所述触发模块,还用于在访问时通过所述接口内的UI自动化程序触发打开自动化部署工具页面,并在所述自动化部署工具页面中创建和自动执行测试任务,所述UI自动化程序中写有用于执行测试的业务逻辑。

[0073] 在一些可能的实施方式中,所述返回模块,还用于通过所述接口内的UI自动化程序定位和获取所述自动化部署工具页面中的输出信息;再通过所述接口获取所述输出信息作为测试返回值。

[0074] 在一些实施例中,本公开实施例提供的装置具有的功能或包含的模块可以用于执行上文方法实施例描述的方法,其具体实现可以参照上文方法实施例的描述,为了简洁,这里不再赘述。

[0075] 此外,本公开还提供了获取测试结果的装置、电子设备、计算机可读存储介质、程序,上述均可用来实现本公开提供的任一种获取测试结果的方法,相应技术方案和描述和参见方法部分的相应记载,不再赘述。

[0076] 本公开实施例还提出一种计算机可读存储介质,其上存储有计算机程序指令,所述计算机程序指令被处理器执行时实现上述方法。计算机可读存储介质可以是非易失性计算机可读存储介质。

[0077] 本公开实施例还提出一种电子设备,包括:处理器;用于存储处理器可执行指令的存储器;其中,所述处理器被配置为上述方法。

[0078] 电子设备可以被提供为终端、服务器或其它形态的设备。

[0079] 图4示出根据本公开实施例的一种电子设备800的框图。例如,电子设备800可以是移动电话,计算机,数字广播终端,消息收发设备,游戏控制台,平板设备,医疗设备,健身设备,个人数字助理等终端。

[0080] 参照图4,电子设备800可以包括以下一个或多个组件:处理组件802,存储器804,电源组件806,多媒体组件808,音频组件810,输入/输出(I/O)的接口812,传感器组件814,以及通信组件816。

[0081] 处理组件802通常控制电子设备800的整体操作,诸如与显示,电话呼叫,数据通信,相机操作和记录操作相关联的操作。处理组件802可以包括一个或多个处理器820来执行指令,以完成上述的方法的全部或部分步骤。此外,处理组件802可以包括一个或多个模块,便于处理组件802和其他组件之间的交互。例如,处理组件802可以包括多媒体模块,以方便多媒体组件808和处理组件802之间的交互。

[0082] 存储器804被配置为存储各种类型的数据以支持在电子设备800的操作。这些数据的示例包括用于在电子设备800上操作的任何应用程序或方法的指令,联系人数据,电话簿数据,消息,图片,视频等。存储器804可以由任何类型的易失性或非易失性存储设备或者它们的组合实现,如静态随机存取存储器(SRAM),电可擦除可编程只读存储器(EEPROM),可擦除可编程只读存储器(EPROM),可编程只读存储器(PROM),只读存储器(ROM),磁存储器,快闪存储器,磁盘或光盘。

[0083] 电源组件806为电子设备800的各种组件提供电力。电源组件806可以包括电源管理系统,一个或多个电源,及其他与为电子设备800生成、管理和分配电力相关联的组件。

[0084] 多媒体组件808包括在所述电子设备800和用户之间的提供一个输出接口的屏幕。在一些实施例中,屏幕可以包括液晶显示器(LCD)和触摸面板(TP)。如果屏幕包括触摸面板,屏幕可以被实现为触摸屏,以接收来自用户的输入信号。触摸面板包括一个或多个触摸传感器以感测触摸、滑动和触摸面板上的手势。所述触摸传感器可以不仅感测触摸或滑动动作的边界,而且还检测与所述触摸或滑动操作相关的持续时间和压力。在一些实施例中,多媒体组件808包括一个前置摄像头和/或后置摄像头。当电子设备800处于操作模式,如拍摄模式或视频模式时,前置摄像头和/或后置摄像头可以接收外部的多媒体数据。每个前置摄像头和后置摄像头可以是一个固定的光学透镜系统或具有焦距和光学变焦能力。

[0085] 音频组件810被配置为输出和/或输入音频信号。例如,音频组件810包括一个麦克风(MIC),当电子设备800处于操作模式,如呼叫模式、记录模式和语音识别模式时,麦克风被配置为接收外部音频信号。所接收的音频信号可以被进一步存储在存储器804或经由通信组件816发送。在一些实施例中,音频组件810还包括一个扬声器,用于输出音频信号。

[0086] I/O接口812为处理组件802和外围接口模块之间提供接口,上述外围接口模块可以是键盘,点击轮,按钮等。这些按钮可包括但不限于:主页按钮、音量按钮、启动按钮和锁定按钮。

[0087] 传感器组件814包括一个或多个传感器,用于为电子设备800提供各个方面的状态评估。例如,传感器组件814可以检测到电子设备800的打开/关闭状态,组件的相对定位,例如所述组件为电子设备800的显示器和小键盘,传感器组件814还可以检测电子设备800或电子设备800一个组件的位置改变,用户与电子设备800接触的存在或不存在,电子设备800方位或加速/减速和电子设备800的温度变化。传感器组件814可以包括接近传感器,被配置用来在没有任何的物理接触时检测附近物体的存在。传感器组件814还可以包括光传感器,如CMOS或CCD图像传感器,用于在成像应用中使用。在一些实施例中,该传感器组件814还可以包括加速度传感器,陀螺仪传感器,磁传感器,压力传感器或温度传感器。

[0088] 通信组件816被配置为便于电子设备800和其他设备之间有线或无线方式的通信。电子设备800可以接入基于通信标准的无线网络,如WiFi,2G或3G,或它们的组合。在一个示例性实施例中,通信组件816经由广播信道接收来自外部广播管理系统的广播信号或广播相关信息。在一个示例性实施例中,所述通信组件816还包括近场通信(NFC)模块,以促进短程通信。例如,在NFC模块可基于射频识别(RFID)技术,红外数据协会(IrDA)技术,超宽带(UWB)技术,蓝牙(BT)技术和其他技术来实现。

[0089] 在示例性实施例中,电子设备800可以被一个或多个应用专用集成电路(ASIC)、数字信号处理器(DSP)、数字信号处理设备(DSPD)、可编程逻辑器件(PLD)、现场可编程门阵列(FPGA)、控制器、微控制器、微处理器或其他电子元件实现,用于执行上述方法。

[0090] 在示例性实施例中,还提供了一种非易失性计算机可读存储介质,例如包括计算机程序指令的存储器804,上述计算机程序指令可由电子设备800的处理器820执行以完成上述方法。

[0091] 图5示出根据本公开实施例的另一种电子设备1900的框图。例如,电子设备1900可以被提供为一服务器。参照图5,电子设备1900包括处理组件1922,其进一步包括一个或多

个处理器,以及由存储器1932所代表的存储器资源,用于存储可由处理组件1922的执行的指令,例如应用程序。存储器1932中存储的应用程序可以包括一个或一个以上的每一个对应于一组指令的模块。此外,处理组件1922被配置为执行指令,以执行上述方法。

[0092] 电子设备1900还可以包括一个电源组件1926被配置为执行电子设备1900的电源管理,一个有线或无线网络接口1950被配置为将电子设备1900连接到网络,和一个输入输出(I/O)接口1958。电子设备1900可以操作基于存储在存储器1932的操作系统,例如Windows Server™,Mac OS X™,Unix™,Linux™,FreeBSD™或类似。

[0093] 在示例性实施例中,还提供了一种非易失性计算机可读存储介质,例如包括计算机程序指令的存储器1932,上述计算机程序指令可由电子设备1900的处理组件1922执行以完成上述方法。

[0094] 本公开可以是系统、方法和/或计算机程序产品。计算机程序产品可以包括计算机可读存储介质,其上载有用于使处理器实现本公开的各个方面的计算机可读程序指令。

[0095] 计算机可读存储介质是可以保持和存储由指令执行设备使用的指令的有形设备。计算机可读存储介质例如可以是一一但不限于一一电存储设备、磁存储设备、光存储设备、电磁存储设备、半导体存储设备或者上述的任意合适的组合。计算机可读存储介质的更具体的例子(非穷举的列表)包括:便携式计算机盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、静态随机存取存储器(SRAM)、便携式压缩盘只读存储器(CD-ROM)、数字多功能盘(DVD)、记忆棒、软盘、机械编码设备、例如其上存储有指令的打孔卡或凹槽内凸起结构、以及上述的任意合适的组合。这里所使用的计算机可读存储介质不被解释为瞬时信号本身,诸如无线电波或者其他自由传播的电磁波、通过波导或其他传输媒介传播的电磁波(例如,通过光纤电缆的光脉冲)、或者通过电线传输的电信号。

[0096] 这里所描述的计算机可读程序指令可以从计算机可读存储介质下载到各个计算/处理设备,或者通过网络、例如因特网、局域网、广域网和/或无线网下载到外部计算机或外部存储设备。网络可以包括铜传输电缆、光纤传输、无线传输、路由器、防火墙、交换机、网关计算机和/或边缘服务器。每个计算/处理设备中的网络适配卡或者网络接口从网络接收计算机可读程序指令,并转发该计算机可读程序指令,以供存储在各个计算/处理设备中的计算机可读存储介质中。

[0097] 用于执行本公开操作的计算机程序指令可以是汇编指令、指令集架构(ISA)指令、机器指令、机器相关指令、微代码、固件指令、状态设置数据、或者以一种或多种编程语言的任意组合编写的源代码或目标代码,所述编程语言包括面向对象的编程语言—诸如Smalltalk、C++等,以及常规的过程式编程语言—诸如“C”语言或类似的编程语言。计算机可读程序指令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)—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)。在一些实施例中,通过利用计算机可读程序指令的状态信息来个性化定制电子电路,例如可编程逻辑电路、现场可编程门阵列(FPGA)或可编程逻辑阵列(PLA),该电子电路可以执行计算机可读程序指令,从而实现本公开的各个方

面。

[0098] 这里参照根据本公开实施例的方法、装置(系统)和计算机程序产品的流程图和/或框图描述了本公开的各个方面。应当理解,流程图和/或框图的每个方框以及流程图和/或框图中各方框的组合,都可以由计算机可读程序指令实现。

[0099] 这些计算机可读程序指令可以提供给通用计算机、专用计算机或其它可编程数据处理装置的处理器,从而生产出一种机器,使得这些指令在通过计算机或其它可编程数据处理装置的处理器执行时,产生了实现流程图和/或框图中的一个或多个方框中规定的功能/动作的装置。也可以把这些计算机可读程序指令存储在计算机可读存储介质中,这些指令使得计算机、可编程数据处理装置和/或其他设备以特定方式工作,从而,存储有指令的计算机可读介质则包括一个制品,其包括实现流程图和/或框图中的一个或多个方框中规定的功能/动作的各个方面的指令。

[0100] 也可以把计算机可读程序指令加载到计算机、其它可编程数据处理装置、或其它设备上,使得在计算机、其它可编程数据处理装置或其它设备上执行一系列操作步骤,以产生计算机实现的过程,从而使得在计算机、其它可编程数据处理装置、或其它设备上执行的指令实现流程图和/或框图中的一个或多个方框中规定的功能/动作。

[0101] 附图中的流程图和框图显示了根据本公开的多个实施例的系统、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、程序段或指令的一部分,所述模块、程序段或指令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。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,两个连续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。也要注意的,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动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。

[0102] 以上已经描述了本公开的各实施例,上述说明是示例性的,并非穷尽性的,并且也不限于所披露的各实施例。在不偏离所说明的各实施例的范围和精神的情况下,对于本技术领域的普通技术人员来说许多修改和变更都是显而易见的。本文中所用术语的选择,旨在最好地解释各实施例的原理、实际应用或对市场中的技术改进,或者使本技术领域的其它普通技术人员能理解本文披露的各实施例。

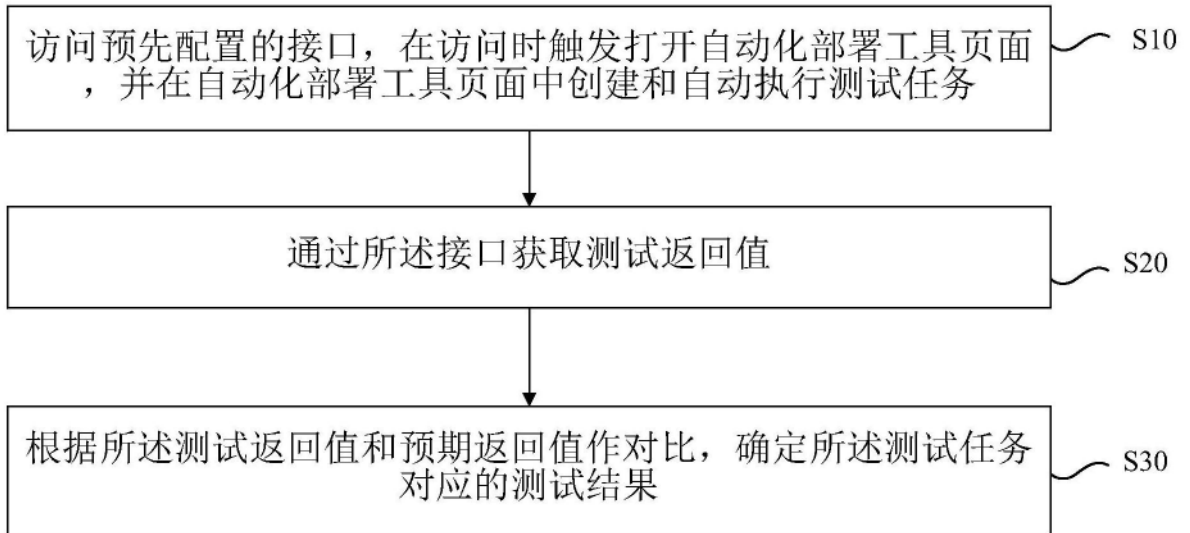


图1

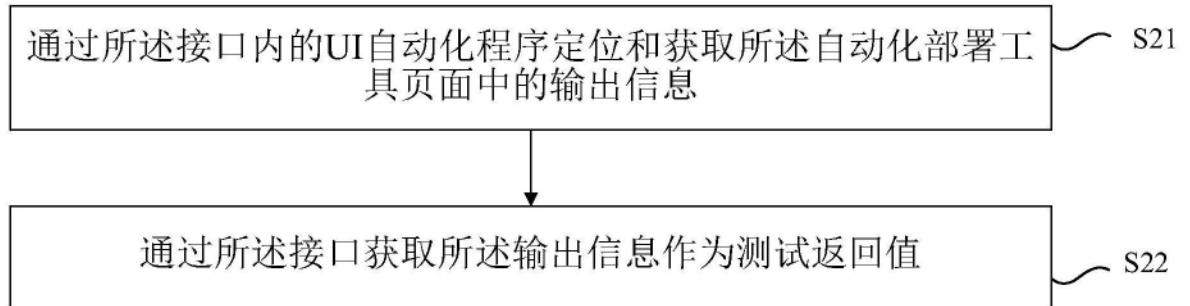


图2

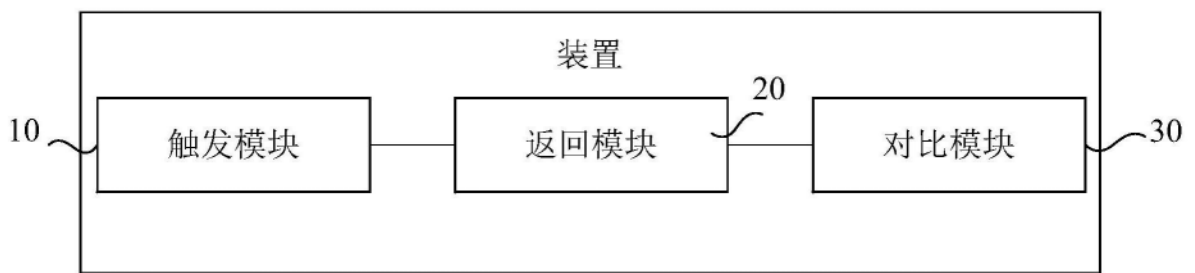


图3

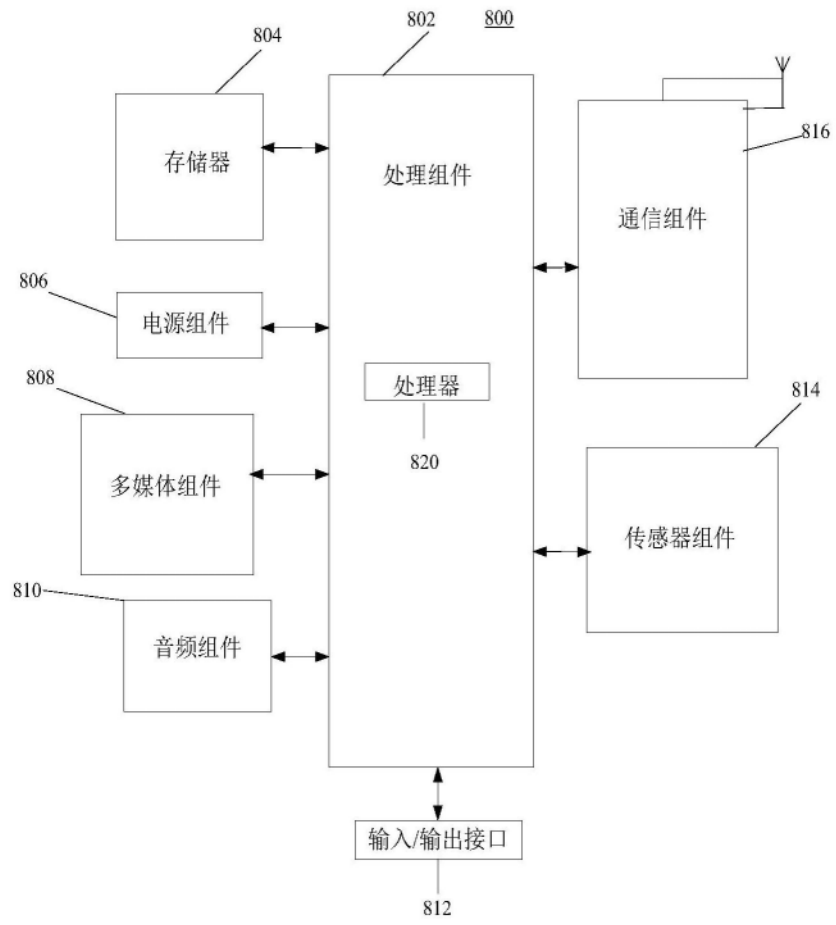


图4

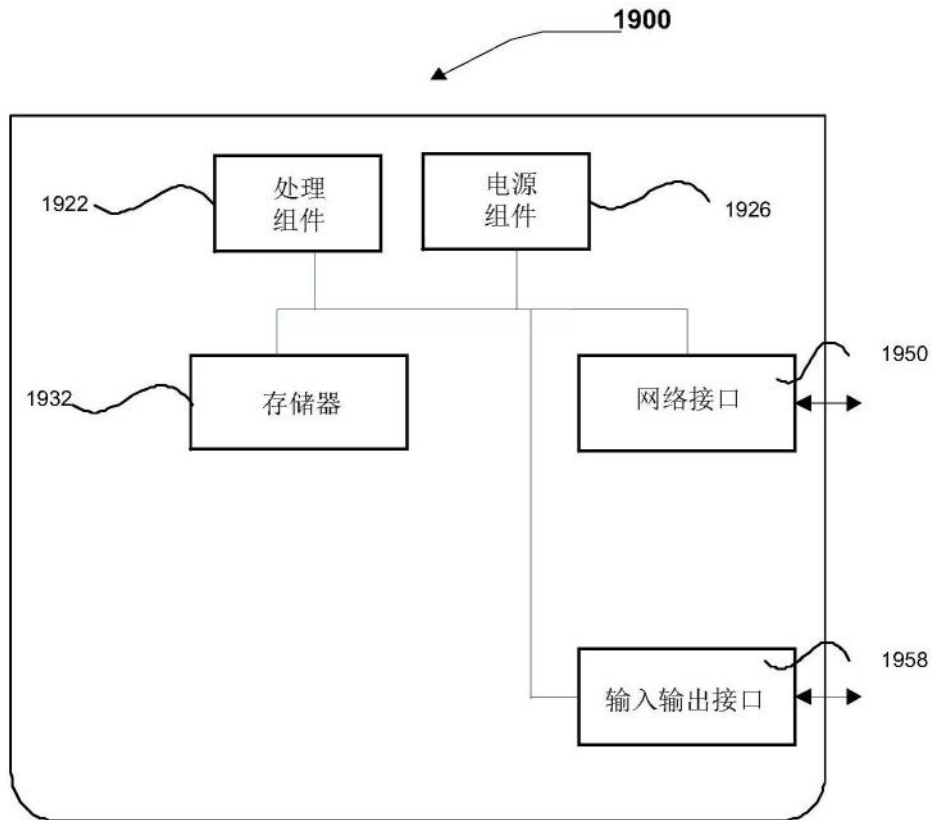


图5