



(12) 发明专利

(10) 授权公告号 CN 108984417 B

(45) 授权公告日 2022.06.03

(21) 申请号 201810931002.0

(22) 申请日 2018.08.15

(65) 同一申请的已公布的文献号

申请公布号 CN 108984417 A

(43) 申请公布日 2018.12.11

(73) 专利权人 北京达佳互联信息技术有限公司

地址 100084 北京市海淀区中关村东路1号
院8号楼20层B2201

(72) 发明人 刘硕

(74) 专利代理机构 北京润泽恒知识产权代理有限公司 11319

专利代理人 莎日娜

(51) Int.Cl.

G06F 11/36 (2006.01)

(56) 对比文件

CN 107948017 A, 2018.04.20

CN 103281366 A, 2013.09.04

CN 107196858 A, 2017.09.22

CN 108255727 A, 2018.07.06

CN 106502915 A, 2017.03.15

CN 108038013 A, 2018.05.15

CN 107231251 A, 2017.10.03

CN 105515872 A, 2016.04.20

CN 106375342 A, 2017.02.01

US 2008276224 A1, 2008.11.06

US 2014122935 A1, 2014.05.01

Yang_ZhiRan.ZooKeeper客户端脚本

zkCli.sh的节点操作.《https://blog.csdn.net/YZR_JAVA/article/details/76136773》.2017,

审查员 杏运

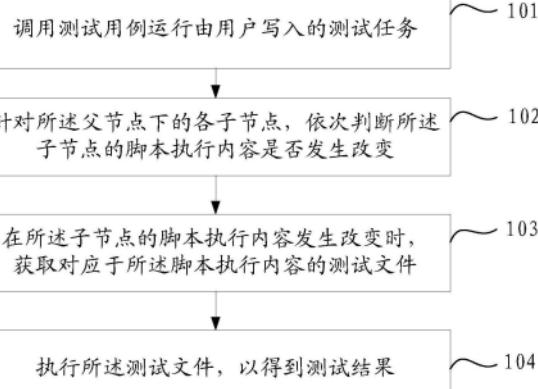
权利要求书2页 说明书10页 附图4页

(54) 发明名称

软件测试方法、装置、终端及存储介质

(57) 摘要

本发明公开了一种软件测试方法、装置、终端及存储介质。所述方法包括：调用测试用例运行由用户写入的测试任务；所述测试用例所在节点为父节点；针对所述父节点下的各子节点，依次判断所述子节点的脚本执行内容是否发生改变；在所述子节点的脚本执行内容发生改变时，获取对应于所述脚本执行内容的测试文件；执行所述测试文件，以得到测试结果。本发明能够保证代码在线上、线下运行都没有问题，因而，能够保证测试用例在测试环境和线上环境运行时返回的结果是一致的。



1. 一种软件测试方法,应用于终端,其特征在于,包括:
调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;
针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;
在所述子节点的脚本执行内容发生改变时,获取对应于所述脚本执行内容的测试文件;
执行所述测试文件,以得到测试结果;
所述获取对应于所述脚本执行内容的测试文件的步骤,包括:
将所述脚本执行内容通过预置接口发送至接收装置;
接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言;
依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。
2. 根据权利要求1所述的方法,其特征在于,在所述调用测试用例运行由用户写入的测试任务的步骤之前,还包括:
确定所述测试用例在Zookeeper中所处的节点;
获取所述测试任务对应的测试代码;
将所述测试代码以脚本的形式写入所述测试用例。
3. 根据权利要求1所述的方法,其特征在于,所述针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变的步骤,包括:
每隔阈值时间,针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变。
4. 根据权利要求1所述的方法,其特征在于,所述预置接口包括统一资源定位符,在所述执行所述测试文件,以得到测试结果的步骤之后,还包括:
依据所述测试结果生成相应的测试报告;
依据所述统一资源定位符信息将所述测试报告发送至目标机房。
5. 一种软件测试装置,其特征在于,包括:
测试任务运行单元,被配置为调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;
脚本内容判断单元,被配置为针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;
测试文件获取单元,被配置为在所述子节点的脚本执行内容发生改变时,获取对应于所述脚本执行内容的测试文件;
测试文件执行单元,被配置为执行所述测试文件,以得到测试结果;
所述测试文件获取单元包括:
脚本内容发送子单元,被配置为将所述脚本执行内容通过预置接口发送至接收装置;
脚本语言获取子单元,被配置为接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言;
脚本内容转换子单元,被配置为依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。
6. 根据权利要求5所述的装置,其特征在于,还包括:
节点确定单元,被配置为确定所述测试用例在Zookeeper中所处的节点;

测试代码获取单元,被配置为获取所述测试任务对应的测试代码;

测试代码写入单元,被配置为将所述测试代码以脚本的形式写入所述测试用例。

7.根据权利要求5所述的装置,其特征在于,所述脚本内容判断单元包括:

脚本内容判断子单元,被配置为每隔阈值时间,针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变。

8.根据权利要求5所述的装置,其特征在于,所述预置接口包括统一资源定位符,还包括:

测试报告生成单元,被配置为依据所述测试结果生成相应的测试报告;

测试报告发送单元,被配置为依据所述统一资源定位符信息将所述测试报告发送至目标机房。

9.一种软件测试的终端,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:

调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;

针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;

如果是,则获取对应于所述脚本执行内容的测试文件;

执行所述测试文件,以得到测试结果;

所述获取对应于所述脚本执行内容的测试文件的步骤,包括:

将所述脚本执行内容通过预置接口发送至接收装置;

接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言;

依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。

10.一种非临时性计算机可读存储介质,当所述存储介质中的指令由终端的处理器执行时,使得终端能够执行一种软件测试的方法,所述方法包括:

调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;

针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;

如果是,则获取对应于所述脚本执行内容的测试文件;

执行所述测试文件,以得到测试结果;

所述获取对应于所述脚本执行内容的测试文件的步骤,包括:

将所述脚本执行内容通过预置接口发送至接收装置;

接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言;

依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。

软件测试方法、装置、终端及存储介质

技术领域

[0001] 本发明涉及计算机技术领域,尤其涉及一种软件测试方法、装置、终端及存储介质。

背景技术

[0002] 随着IT技术的发展,软件覆盖的行业范围越来越广,系统也变得越得越来越复杂。

[0003] 针对各种复杂的软件系统,在实际生产过程中,线上代码的运行环境同线下测试的运行环境总会存在差别,而怎样有效的开展软件测试,以保证代码在线上、线下运行都没问题,且测试用例在测试环境和线上环境运行时要返回一样的结果,是软件行业目前关注的问题。

发明内容

[0004] 为克服相关技术中存在的问题,本发明提供一种软件测试方法、装置、终端及存储介质。

[0005] 根据本发明实施例的第一方面,提供一种软件测试方法,包括:调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;在所述子节点的脚本执行内容发生改变时,获取对应于所述脚本执行内容的测试文件;执行所述测试文件,以得到测试结果。

[0006] 优选地,在所述调用测试用例运行由用户写入的测试任务的步骤之前,还包括:确定所述测试用例在Zookeeper中所处的节点;获取所述测试任务对应的测试代码;将所述测试代码以脚本的形式写入所述测试用例。

[0007] 优选地,所述针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变的步骤,包括:每隔阈值时间,针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变。

[0008] 优选地,所述获取对应于所述脚本执行内容的测试文件的步骤,包括:将所述脚本执行内容通过预置接口发送至接收装置;接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言;依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。

[0009] 优选地,所述预置接口包括统一资源定位符,在所述执行所述测试文件,以得到测试结果的步骤之后,还包括:依据所述测试结果生成相应的测试报告;依据所述统一资源定位符信息将所述测试报告发送至目标机房。

[0010] 根据本发明实施例的第二方面,提供一种软件测试装置,包括:测试任务运行单元,被配置为调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;脚本内容判断单元,被配置为针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;测试文件获取单元,被配置为在所述子节点的脚本执行内容发生改变时,获取对应于所述脚本执行内容的测试文件;测试文件执行单元,被配置为执行所述

测试文件,以得到测试结果。

[0011] 优选地,还包括:节点确定单元,被配置为确定所述测试用例在Zookeeper中所处的节点;测试代码获取单元,被配置为获取所述测试任务对应的测试代码;测试代码写入单元,被配置为将所述测试代码以脚本的形式写入所述测试用例。

[0012] 优选地,所述脚本内容判断单元包括:脚本内容判断子单元,被配置为每隔阈值时间,针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变。

[0013] 优选地,所述测试文件获取单元包括:脚本内容发送子单元,被配置为将所述脚本执行内容通过预置接口发送至接收装置;脚本语言获取子单元,被配置为接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言;脚本内容转换子单元,被配置为依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。

[0014] 优选地,还包括:测试报告生成单元,被配置为依据所述测试结果生成相应的测试报告;测试报告发送单元,被配置为依据所述统一资源定位符信息将所述测试报告发送至目标机房。

[0015] 根据本发明实施例的第三方面,提供了一种软件测试的终端,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:调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;如果是,则获取对应于所述脚本执行内容的测试文件;执行所述测试文件,以得到测试结果。

[0016] 根据本发明实施例的第四方面,提供了一种非临时性计算机可读存储介质,当所述存储介质中的指令由终端的处理器执行时,使得终端能够执行一种软件测试的方法,所述方法包括:调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;如果是,则获取对应于所述脚本执行内容的测试文件;执行所述测试文件,以得到测试结果。

[0017] 本发明的实施例提供的技术方案可以包括以下有益效果:

[0018] 本发明实施例提供的软件测试方案,通过调用测试用例运行写入的测试任务,该测试用例所在节点为父节点,针对该父节点下的各子节点,依次判断该子节点的脚本执行内容是否发生改变,在该子节点的脚本执行内容发生改变时,则获取对应于该脚本执行内容的测试文件进行测试,以得到测试结果。由于本发明实施例提供的软件测试方案,能够对测试任务所处父节点下的各子节点的脚本内容是否发生改变进行监测,因此,本发明能够保证代码在线上、线下运行都没有问题,因而,能够保证测试用例在测试环境和线上环境运行时返回的结果是一致的。

[0019] 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,并不能限制本发明。

附图说明

[0020] 此处的附图被并入说明书中并构成本说明书的一部分,示出了符合本发明的实施例,并与说明书一起用于解释本发明的原理。

[0021] 图1是根据本发明实施例一的一种软件测试方法的步骤流程图。

[0022] 图2是根据本发明实施例二的一种软件测试方法的步骤流程图

- [0023] 图3是根据本发明实施例三的一种软件测试装置的结构框图。
- [0024] 图4是根据本发明实施例四的一种软件测试的终端的结构框图。

具体实施方式

[0025] 这里将详细地对示例性实施例进行说明,其示例表示在附图中。下面的描述涉及附图时,除非另有表示,不同附图中的相同数字表示相同或相似的要素。以下示例性实施例中所描述的实施方式并不代表与本发明相一致的所有实施方式。相反,它们仅是与如所附权利要求书中所详述的、本发明的一些方面相一致的装置和方法的例子。

- [0026] 实施例一

[0027] 参照图1,示出了本发明实施例提供的一种软件测试方法的流程图,该软件测试方法可以应用于终端中,具体可以包括如下步骤:

- [0028] 步骤101:调用测试用例由用户运行写入的测试任务。

[0029] 在本发明实施例中,终端可以包括计算机、PAD (Portable Android Device,平板电脑) 等电子设备。

[0030] 测试用例 (Test Case) 是指为某个特殊目标而编制的一组测试输入、执行条件以及预期结果,以便测试某个程序路径或核实是否满足某个特定需求。

[0031] 在终端系统中预先设置有测试用例,该测试用例所处的节点为父节点,在父节点下还可以包含一个或多个子节点,如1个、5个、10个等等,本发明对此不加以限制。

[0032] 在研发人员将测试任务写入测试用例之后,可以调用该测试用例运行写入的测试任务。

- [0033] 在调用测试用例运行写入的测试任务之后,执行步骤102。

[0034] 步骤102:针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变。

[0035] 在测试任务写入之后,可以调用测试用例运行该测试任务,进而测试任务可以被分为多个子任务,并被分配至测试用例所处父节点下的各个子节点运行。

[0036] 针对父节点下的各子节点,可以依次判断各子节点的脚本执行内容是否发生变化,例如,脚本内容数据发生改变,脚本内容格式发生改变等等。

[0037] 在父节点下的某子节点脚本执行内容未发生改变时,则继续对该子节点的脚本执行内容进行检测。

- [0038] 在父节点下的某子节点脚本执行内容发生改变时,则执行步骤103。

[0039] 步骤103:在所述子节点的脚本执行内容发生改变时,获取对应于所述脚本执行内容的测试文件。

[0040] 在某一节点的脚本执行内容发生改变时,可以依据脚本执行内容获取脚本执行内容对应的测试文件,例如,依据脚本执行内容的脚本语言生成对应的测试文件等等,具体地,将在下述实施例二中进行详细描述,在此不再加以赘述。

- [0041] 在获取对应于脚本执行内容的测试文件之后,执行步骤104。

- [0042] 步骤104:执行所述测试文件,以得到测试结果。

[0043] 在生成对应于脚本执行内容的测试文件之后,可以执行该测试文件,从而可以得到测试结果,从测试结果中可以反映出在软件运行过程中具体是哪些内容发生了改变,可

以由研发人员及时掌握软件在线上运行的状况。

[0044] 本发明实施例提供的软件测试方法，调用测试用例运行由用户写入的测试任务，该测试用例所在节点为父节点，针对该父节点下的各子节点，依次判断该子节点的脚本执行内容是否发生改变，在该子节点的脚本执行内容发生改变时，则获取对应于该脚本执行内容的测试文件进行测试，以得到测试结果。由于本发明实施例提供的软件测试方案，能够对测试任务所处父节点下的各子节点的脚本内容是否发生改变进行监测，因此，本发明能够保证代码在线上、线下运行都没有问题，因而，能够保证测试用例在测试环境和线上环境运行时返回的结果是一致的。

[0045] 实施例二

[0046] 参照图2，示出了本发明实施例提供的一种软件测试方法的流程图，该软件测试方法可以应用于终端中，具体可以包括如下步骤：

[0047] 步骤201：确定所述测试用例在Zookeeper中所处的节点。

[0048] 在本发明实施例中，终端可以包括计算机、PAD (Portable Android Device，平板电脑) 等电子设备。

[0049] ZooKeeper是一个分布式的，开放源码的分布式应用程序协调服务，ZooKeeper可以预先部署父节点和隶属于该父节点的子节点，并在父节点设置对应的测试用例，即测试用例在Zookeeper中所处的节点为父节点。

[0050] 测试用例所处节点的路径和具体的业务服务器在同一个父亲节点内，例如，测试用例所处节点的路径是/zk/biz1/tests/case1,/zk/biz1/tests/case2…，业务服务器的路径分别是/zk/biz1/hosts/server1,/zk/biz/hosts/server2…，表示二者在同一个业务/zk/biz1下。进而，系统中预置的探测程序可以依据某一个业务对应的服务器地址以及需要运行测试用例脚本信息，确定测试用例所处的节点。

[0051] 在确定测试用例在Zookeeper中所处的节点之后，执行步骤202。

[0052] 步骤202：获取所述测试任务对应的测试代码。

[0053] 测试任务是由研发人员通过代码的形式写入的，在系统确定研发人员写入的测试任务之后，即可获取到对应的测试代码。

[0054] 在获取到测试任务对应的测试代码后，执行步骤203。

[0055] 步骤203：将所述测试代码以脚本的形式写入所述测试用例。

[0056] 脚本是批处理文件的延伸，是一种纯文本保存的程序。在获取到测试任务对应的代码后，可以将测试代码以脚本的形式写入测试用例，并执行步骤204。

[0057] 步骤204：调用测试用例运行由用户写入的测试任务。

[0058] 在将测试任务写入测试用例之后，可以调用该测试用例运行写入的测试任务，并执行步骤205。

[0059] 步骤205：每隔阈值时间，针对所述父节点下的各子节点，依次判断所述子节点的脚本执行内容是否发生改变。

[0060] 在本发明实施例中，阈值时间可以为1min、3min、8min等等，本发明对此不加以限制。在终端系统中可以预先设置定时器，在达到定时器设置的时间时，HTTP探测服务器可以对父节点下的各子节点进行一次轮询操作，以判断各子节点的脚本执行内容是否发生了改变。

[0061] 可以通过脚本执行内容的数据是否发生改变(如数据增加或减少等等),脚本执行内容的格式是否发生改变等方式来确定各子节点的脚本执行内容是否发生改变。

[0062] 在具体实现中,还可以采用其它方式来判断各子节点的脚本执行内容是否发生改变,本发明实施例对此不加以限制。

[0063] 在父节点下的某子节点脚本执行内容未发生改变时,则继续对该子节点的脚本执行内容进行检测。

[0064] 在父节点下的某子节点脚本执行内容发生改变时,则执行步骤206。

[0065] 步骤206:将所述脚本执行内容通过预置接口发送至接收装置。

[0066] 在终端本地预先设置有接收装置,如agent接收装置等。在检测到某子节点的脚本执行内容发生改变后,可以通过预置接口(如HTTP接口等)将发生改变的脚本执行内容发送至接收装置。

[0067] 在将脚本执行内容发送至接收装置的同时,还可以将脚本执行内容所处节点的标识发送至接收装置,例如,脚本执行内容所处的节点标识为A,可以将节点标识A与脚本执行内容进行关联,一同发送至接收装置,从而可以识别出哪些节点的脚本执行内容发生了改变。

[0068] 在将脚本执行内容发送至接收装置之后,执行步骤207。

[0069] 步骤207:接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言。

[0070] 接收装置可以根据脚本执行内容获取对应的脚本语言,如python、ruby、node j等语言。在接收装置获取脚本执行内容对应的脚本语言,可以接收由接收装置发送的脚本执行内容对应的脚本语言及对应的节点标识,从而执行步骤208。

[0071] 步骤208:依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。

[0072] 在获取脚本执行内容对应的脚本语言之后,可以依据脚本语言对脚本执行内容进行格式转换,以生成对应的测试文件,例如,脚本执行内容对应的脚本语言为.java,则可以将脚本执行内容的格式转换为.java格式等等。

[0073] 可以理解地,上述示例仅是为了更好地理解本发明实施例的技术方案而列举的示例,不作为对本发明的唯一限制。

[0074] 在生成对应的测试文件之后,执行步骤209。

[0075] 步骤209:执行所述测试文件,以得到测试结果。

[0076] 在生成对应于脚本执行内容的测试文件之后,可以执行该测试文件,从而可以得到测试结果,从测试结果中可以反映出在软件运行过程中具体是哪些内容发生了改变,可以由研发人员及时掌握软件在线上运行的状况。

[0077] 在本发明实施例中,可以通过生成日志的方式来保存测试结果,研发人员可以通过实时查看日志的方式来掌握软件的线上运行状况。

[0078] 当然,在实际应用中,也可以采用其它方式保存测试结果,本发明实施例对此不加以限制。

[0079] 步骤210:依据所述测试结果生成相应的测试报告。

[0080] 在得到测试结果之后,可以依据测试结果生成对应的测试报告,可以理解地,在系统中,测试结果通常是以日志的形式保存的,而日志对于非专业人员来说是不容易看懂的,

因此,在本发明实施例中,可以依据测试结果生成相应的测试报告,例如,将日志形式的测试结果转换为文字形式的测试报告等。

[0081] 在生成与测试结果对应的测试报告之后,执行步骤211。

[0082] 步骤211:依据所述统一资源定位符信息将所述测试报告发送至目标机房。

[0083] 在终端系统中,预先保存有各机房对应的统一资源定位符信息(URL,Uniform Resource Locator),即网页地址。

[0084] 目标机房是指运行测试任务对应的软件的机房,目标机房的个数可以为一个,也可以为多个,具体地,可以根据实际情况下的软件运行情况而定,本发明实施例对此不加以限制。

[0085] 在生成测试报告之后,可以将测试报告分发给目标机房,以由目标机房可以实时了解掌握软件的运行状况。

[0086] 本发明实施例提供的软件测试方法,除了具有上述实施例一所述的软件测试方法所具有的有益效果外,还可以依据测试结果生成测试报告,以发送给目标机房,从而使得相关人员能够实时了解掌握软件的运行状况,极大地提升了用户的使用体验。

[0087] 实施例三

[0088] 参照图3,示出了本发明实施例提供的一种软件测试装置的结构框图。

[0089] 本发明实施例提供的软件测试装置可以包括:测试任务运行单元301,被配置为调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;脚本内容判断单元302,被配置为针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;测试文件获取单元303,被配置为在所述子节点的脚本执行内容发生改变时,获取对应于所述脚本执行内容的测试文件;测试文件执行单元304,被配置为执行所述测试文件,以得到测试结果。

[0090] 在一个可选的实现方式中,还包括:节点确定单元,被配置为确定所述测试用例在Zookeeper中所处的节点;测试代码获取单元,被配置为获取所述测试任务对应的测试代码;测试代码写入单元,被配置为将所述测试代码以脚本的形式写入所述测试用例所处的节点。

[0091] 在一个可选的实现方式中,所述脚本内容判断单元302包括:脚本内容判断子单元,被配置为每隔阈值时间,针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变。

[0092] 在一个可选的实现方式中,所述测试文件获取单元303包括:脚本内容发送子单元,被配置为将所述脚本执行内容通过预置接口发送至接收装置;脚本语言获取子单元,被配置为接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言;脚本内容转换子单元,被配置为依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。

[0093] 在一个可选的实现方式中,还包括:测试报告生成单元,被配置为依据所述测试结果生成相应的测试报告;测试报告发送单元,被配置为依据所述统一资源定位符信息将所述测试报告发送至目标机房。

[0094] 本发明实施例提供的软件测试装置,调用测试用例运行写入的测试任务,该测试用例所在节点为父节点,针对该父节点下的各子节点,依次判断该子节点的脚本执行内容

是否发生改变，在该子节点的脚本执行内容发生改变时，则获取对应于该脚本执行内容的测试文件进行测试，以得到测试结果。由于本发明实施例提供的软件测试方案，能够对测试任务所处父节点下的各子节点的脚本内容是否发生改变进行监测，因此，本发明能够保证代码在线上、线下运行都没有问题，因而，能够保证测试用例在测试环境和线上环境运行时返回的结果是一致的。

[0095] 实施例四

[0096] 图4是根据一示例性实施例示出的一种用于软件测试的终端800的框图。例如，终端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。

[0097] 参照图4，终端800可以包括以下一个或多个组件：处理组件802，存储器804，电力组件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。

[0098] 处理组件802通常控制终端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。

[0099] 存储器804被配置为存储各种类型的数据以支持在终端800的操作。这些数据的示例包括用于在终端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。

[0100] 电源组件806为终端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，及其他与为终端800生成、管理和分配电力相关联的组件。

[0101] 多媒体组件808包括在所述终端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。

[0102] 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)，当终端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。

[0103] 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。

[0104] 传感器组件814包括一个或多个传感器,用于为终端800提供各个方面状态评估。例如,传感器组件814可以检测到终端800的打开/关闭状态,组件的相对定位,例如所述组件为装置800的显示器和小键盘,传感器组件814还可以检测终端800或终端800一个组件的位置改变,用户与终端800接触的存在或不存在,装置800方位或加速/减速和终端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。

[0105] 通信组件816被配置为便于终端800和其他设备之间有线或无线方式的通信。终端800可以接入基于通信标准的无线网络,如WiFi,运营商网络(如2G、3G、4G或5G),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。

[0106] 在示例性实施例中,终端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,用于执行软件测试方法,具体地,软件测试方法包括:调用测试用例运行由用户写入的测试任务;所述测试用例所在节点为父节点;针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变;在所述子节点的脚本执行内容发生改变时,获取对应于所述脚本执行内容的测试文件;执行所述测试文件,以得到测试结果。

[0107] 可选地,在所述调用测试用例运行由用户写入的测试任务的步骤之前,还包括:确定所述测试用例在Zookeeper中所处的节点;获取所述测试任务对应的测试代码;将所述测试代码以脚本的形式写入所述测试用例。

[0108] 可选地,所述针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变的步骤,包括:每隔阈值时间,针对所述父节点下的各子节点,依次判断所述子节点的脚本执行内容是否发生改变。

[0109] 可选地,所述获取对应于所述脚本执行内容的测试文件的步骤,包括:将所述脚本执行内容通过预置接口发送至接收装置;接收由所述接收装置根据所述脚本执行内容获取的对应的脚本语言;依据所述脚本语言对所述脚本执行内容进行转换,以生成对应的所述测试文件。

[0110] 可选地,所述预置接口包括统一资源定位符,在所述执行所述测试文件,以得到测试结果的步骤之后,还包括:依据所述测试结果生成相应的测试报告;依据所述统一资源定位符信息将所述测试报告发送至目标机房。

[0111] 在示例性实施例中,还提供了一种包括指令的非临时性计算机可读存储介质,例如包括指令的存储器804,上述指令可由终端800的处理器820执行以完成上述方法。例如,所述非临时性计算机可读存储介质可以是ROM、随机存取存储器(RAM)、CD-ROM、磁带、软盘和光数据存储设备等。

[0112] 一种非临时性计算机可读存储介质，当存储介质中的指令由终端的处理器执行时，使得移动终端能够执行本发明中所述的任意一种软件测试方法的步骤。

[0113] 本发明实施例提供的终端，通过调用测试用例运行写入的测试任务，该测试用例所在节点为父节点，针对该父节点下的各子节点，依次判断该子节点的脚本执行内容是否发生改变，若该子节点的脚本执行内容发生该，则获取对应于该脚本执行内容的测试文件进行测试，以得到测试结果。由于本发明实施例提供的软件测试方案，能够对测试任务所处父节点下的各子节点的脚本内容是否发生改变进行监测，因此，本发明能够保证代码在线上、线下运行都没有问题，因而，能够保证测试用例在测试环境和线上环境运行时返回的结果是一致的。

[0114] 对于装置实施例而言，由于其与方法实施例基本相似，所以描述的比较简单，相关之处参见方法实施例的部分说明即可。

[0115] 在此提供的软件测试方案不与任何特定计算机、虚拟系统或者其它设备固有相关。各种通用系统也可以与基于在此的示教一起使用。根据上面的描述，构造具有本发明方案的系统所要求的结构是显而易见的。此外，本发明也不针对任何特定编程语言。应当明白，可以利用各种编程语言实现在此描述的本发明的内容，并且上面对特定语言所做的描述是为了披露本发明的最佳实施方式。

[0116] 在此处所提供的说明书中，说明了大量具体细节。然而，能够理解，本发明的实施例可以在没有这些具体细节的情况下实践。在一些实例中，并未详细示出公知的方法、结构和技术，以便不模糊对本说明书的理解。

[0117] 类似地，应当理解，为了精简本公开并帮助理解各个发明方面中的一个或多个，在上面对本发明的示例性实施例的描述中，本发明的各个特征有时被一起分组到单个实施例、图、或者对其的描述中。然而，并不应将该公开的方法解释成反映如下意图：即所要求保护的本发明要求比在每个权利要求中所明确记载的特征更多的特征。更确切地说，如权利要求书所反映的那样，发明方面在于少于前面公开的单个实施例的所有特征。因此，遵循具体实施方式的权利要求书由此明确地并入该具体实施方式，其中每个权利要求本身都作为本发明的单独实施例。

[0118] 本领域那些技术人员可以理解，可以对实施例中的设备中的模块进行自适应性地改变并且把它们设置在与该实施例不同的一个或多个设备中。可以把实施例中的模块或单元或组件组合成一个模块或单元或组件，以及此外可以把它们分成多个子模块或子单元或子组件。除了这样的特征和/或过程或者单元中的至少一些是相互排斥之外，可以采用任何组合对本说明书(包括伴随的权利要求、摘要和附图)中公开的所有特征以及如此公开的任何方法或者设备的所有过程或单元进行组合。除非另外明确陈述，本说明书(包括伴随的权利要求、摘要和附图)中公开的每个特征可以由提供相同、等同或相似目的的替代特征来代替。

[0119] 此外，本领域的技术人员能够理解，尽管在此所述的一些实施例包括其它实施例中所包括的某些特征而不是其它特征，但是不同实施例的特征的组合意味着处于本发明的范围之内并且形成不同的实施例。例如，在权利要求书中，所要求保护的实施例的任意之一都可以以任意的组合方式来使用。

[0120] 本发明的各个部件实施例可以以硬件实现，或者以在一个或者多个处理器上运行

的软件模块实现，或者以它们的组合实现。本领域的技术人员应当理解，可以在实践中使用微处理器或者数字信号处理器 (DSP) 来实现根据本发明实施例的视频文件转码方案中的一些或者全部部件的一些或者全部功能。本发明还可以实现为用于执行这里所描述的方法的一部分或者全部的设备或者装置程序(例如，计算机程序和计算机程序产品)。这样的实现本发明的程序可以存储在计算机可读介质上，或者可以具有一个或者多个信号的形式。这样的信号可以从因特网网站上下载得到，或者在载体信号上提供，或者以任何其他形式提供。

[0121] 应该注意的是上述实施例对本发明进行说明而不是对本发明进行限制，并且本领域技术人员在不脱离所附权利要求的范围的情况下可设计出替换实施例。在权利要求中，不应将位于括号之间的任何参考符号构造成对权利要求的限制。单词“包含”不排除存在未列在权利要求中的元件或步骤。位于元件之前的单词“一”或“一个”不排除存在多个这样的元件。本发明可以借助于包括有若干不同元件的硬件以及借助于适当编程的计算机来实现。在列举了若干装置的单元权利要求中，这些装置中的若干个可以是通过同一个硬件项来具体体现。单词第一、第二、以及第三等的使用不表示任何顺序。可将这些单词解释为名称。

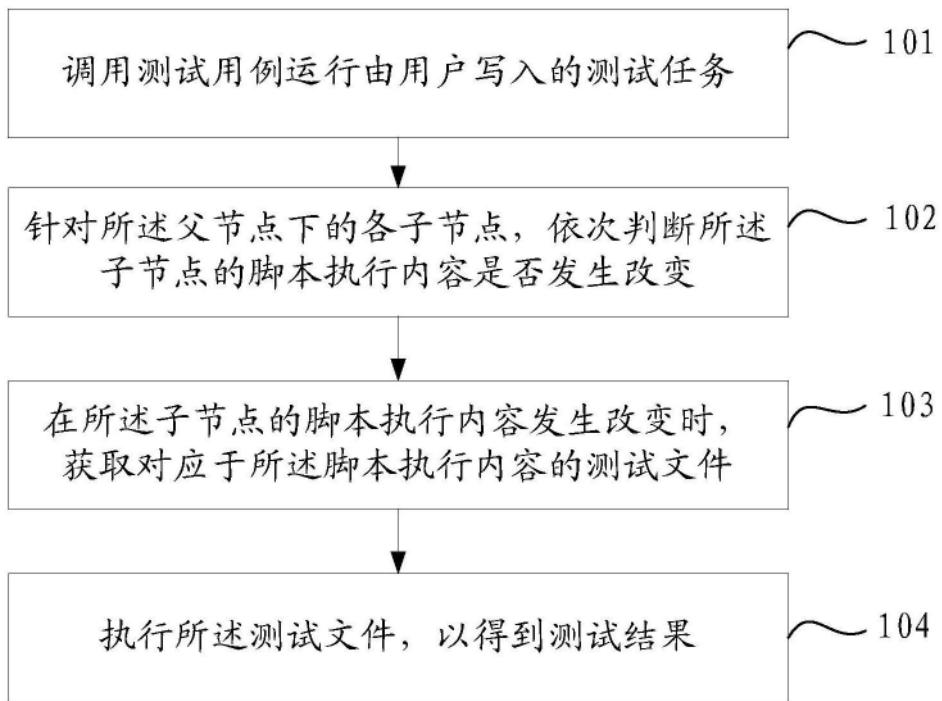


图1

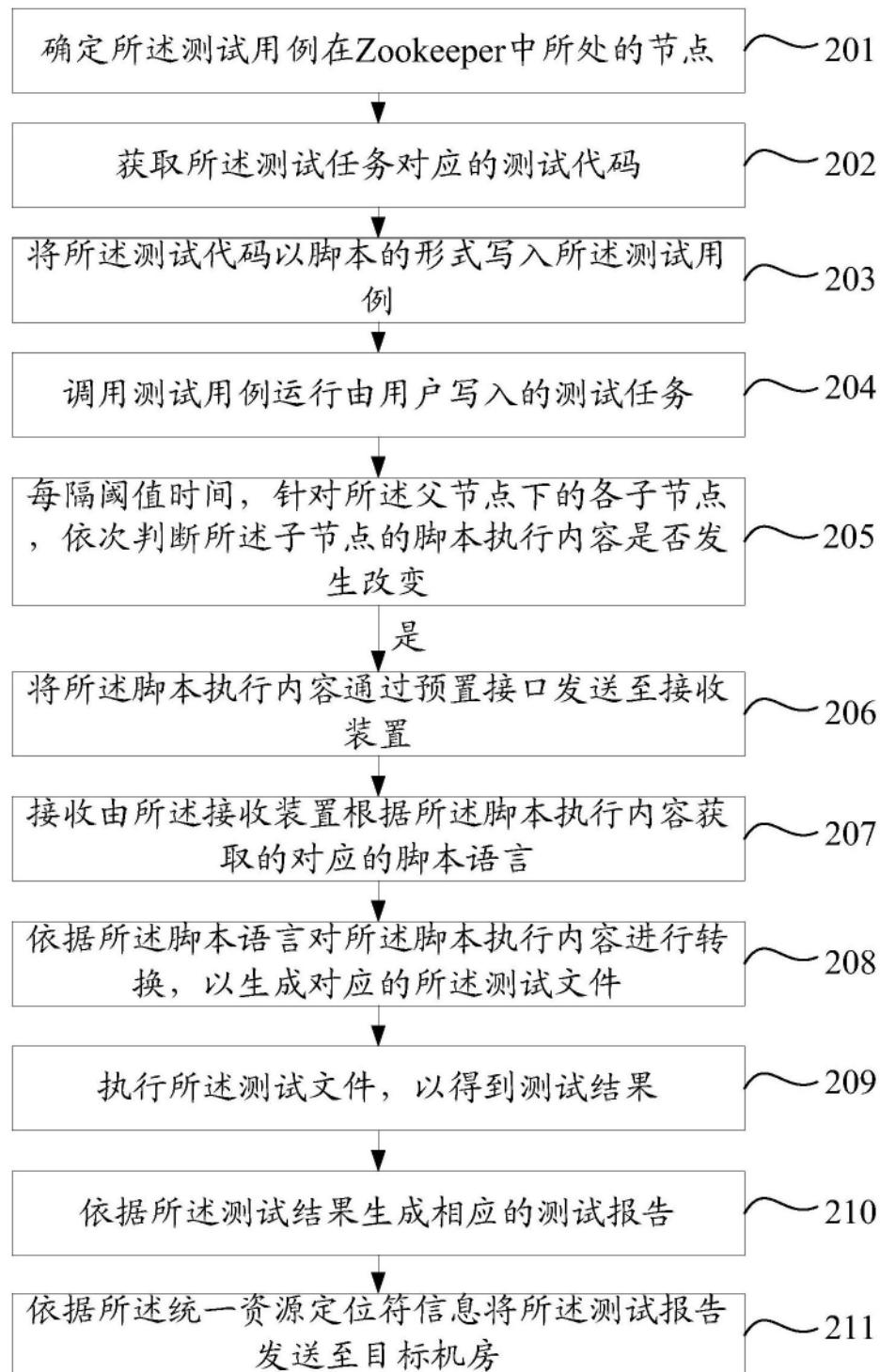


图2

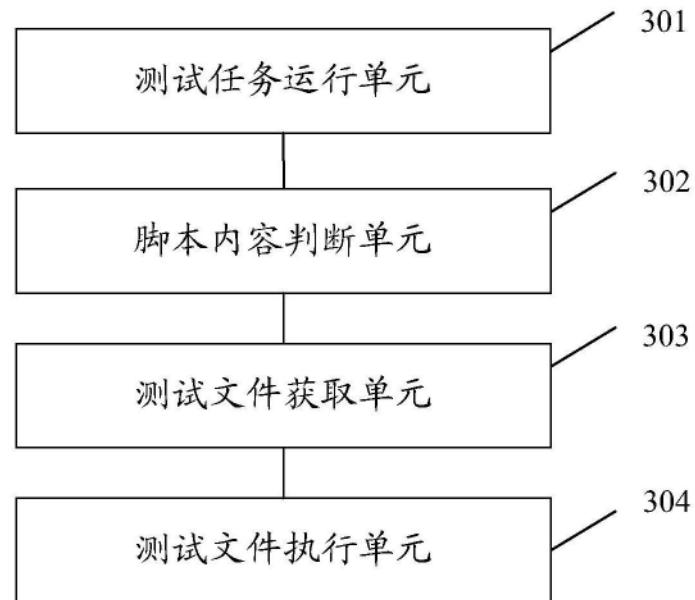


图3

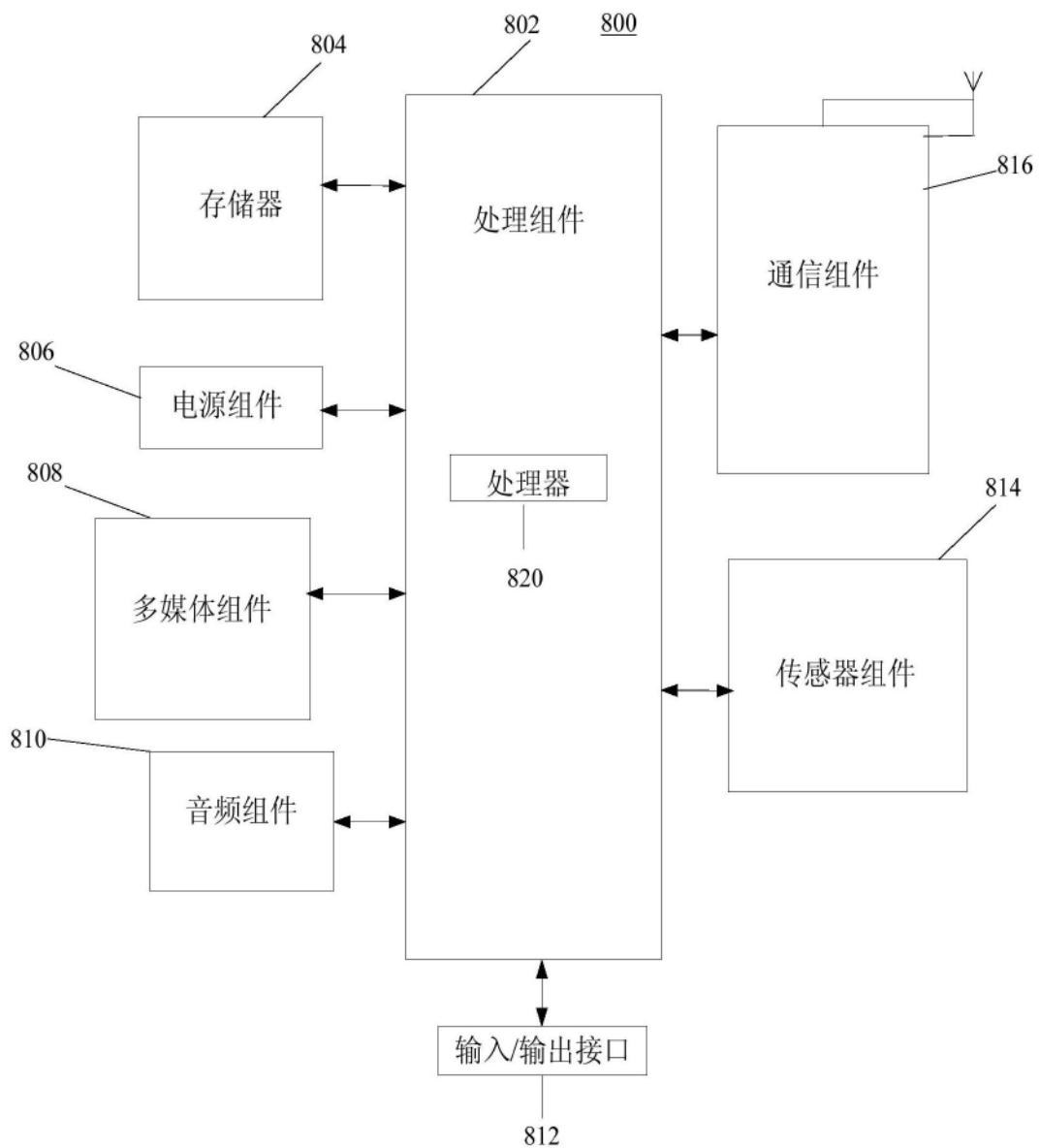


图4