

GUI 自动化测试工具在民用航空数据链系统集成中的应用^①

崔诗娴, 陈春晓, 宫伟祥

(中电科航空电子, 成都 611731)

摘要: 随着综合化、模块化的民用航空电子系统的高速发展, 以及航空数据链系统的广泛应用, 航空数据链系统集成自动化测试技术需求越发迫切. GUI 自动化测试工具是一种基于友好人机界面的软件测试工具, 本文结合具体应用需求, 提出一种改进的 GUI 自动化测试框架, 采用 HMI(Human Machine Interface)界面, 将其应用于民用航空数据链系统中, 将软件内部模块手动测试转换为功能性的黑盒自动化测试, 提高了测试效率.

关键词: 数据链系统; GUI; 自动化测试; 航空电子; HMI; test harness

Application of GUI Automatic Test Tool in Civil Aircraft Data Link System

CUI Shi-Xian, CHEN Chun-Xiao, GONG Wei-Xiang

(China Electronics Technology Group Avionics Corporation(CETCA), Chengdu 611731, China)

Abstract: With the rapid development of the integrated and modularized civil aircraft electronic system and the widely use of aircraft data link system, the requirement of the automatic test in aircraft data link system becomes imperative. GUI automatic test is based on HMI. This document introduces an improved GUI automatic test framework based on HMI (Human Machine Interface) which is implemented in civil aircraft data link system. It changes the manual test of internal modules into the automatic test of the functional black box which makes the test more efficient.

Key words: datalink system; GUI; automatic test; civil aircraft electronic; HMI; test harness

ATS 和 CSA 是两款重要的民用航空机载数据链软件, 其开发方式均为迭代式开发, 需要根据各个里程碑节点完成相应的软件功能. 对这两款软件进行 GUI 测试, 通常的方式是手动测试, 测试人员根据测试用例逐个测试, 人力完成各个检查点的验证, 以此完成常规测试, 回归测试以及覆盖率测试等测试任务^[1].

数据链项目的 GUI 测试和传统意义上的 GUI 测试有所不同, 传统的 GUI 测试主要测试界面的功能和稳定性. 而数据链项目的测试不仅需要测试界面的功能, 并且重点的是通过操作 HMI 模拟器软件, 测试驻留在 IMA 上软件的功能和稳定性.

因此, 鉴于基于数据链的软件需求相对稳定, 测试脚本重复率较高, 为了减少人力成本, 提高测试效

率, 需要进行自动化测试. 一方面, 自动化测试可以在已有功能的回归测试和覆盖率测试上, 减少大量的人力和重复工作; 同时, 自动化脚本就决定了每次执行的测试用例, 测试用例和测试规程均保持一致, 可以减少人为主观性导致的测试用例执行差异.

1 GUI 自动化测试的发展

目前, 国际上通行的 GUI 自动化测试工具主要有 UFT(Unified Functional Testing), RFT(Rational Functional Tester), Test Partner 等, 这些工具具有的特点包括: (1)测试 PC 上所有的模拟器软件进行操作, (2)插入各种检查点, (3)测试结果的截取, (4)产生自定义测试结果报告, (5)成熟的自动化测试框架

^① 基金项目:大数据一体机研发及产业化(2015GZ0111)

收稿时间:2015-11-09;收到修改稿时间:2015-12-24 [doi:10.15888/j.cnki.csa.005178]

和函数库等^[2]。

一般来说, GUI 自动化测试工具需要满足:

(1)支持脚本化语言,多数 GUI 测试工具都支持 VB, C 或者 Java 等通用语言。

(2)录制脚本功能,通过一次人力操作,将操作行为记录下来,自动生成自动化脚本。

(3)有非常高的程序界面对象识别能力,录制的测试脚本才具有更好的可读性,灵活性和更大的修改空间。

(4)支持函数的可重用并支持外部函数库,可以生成私有函数库,也可以调用外部函数库,大大减少工作量。

(5)建立脚本执行任务,批量执行,批量处理和记录测试结果^[3]。

然而,针对 DO-178C 对软件需求的具体要求,ATS 为 C 级软件,CSA 为 D 级软件,在实施软件设计时,统一将软件需求分为了高级需求和低级需求,高级需求追踪到系统需求,低级需求追踪到高级需求,而目标是代码则需要追踪到低级需求,低级需求是指导程序员进行编码的直接引导,而基于软件低级需求开发的测试用例在使用以上工具进行测试时,通常一半以上不能开展自动化测试。现有的 GUI 自动化测试工具只是把 GUI 界面当作黑盒,对其进行输入操作和输出结果读取^[4]。而在 ATS 和 CSA 等数据链软件测试中,GUI 自动化测试工具不仅要把 HMI 软件当作黑盒进行测试,还要根据不同级别的测试策略,将 application, module 和 block 当作黑盒进行测试。因此,我们必须开发出适合此要求的新的 GUI 自动化测试工具。

2 GUI 自动化测试解决方案

2.1 一种新的 GUI 自动化测试框架

在现有的 GUI 自动化测试工具的使用上,我们开发了开发一套新的(additional developed software)优化软件,将其与现有的 GUI 测试工具结合起来,形成新的 GUI 自动化测试框架。优化软件物理上内嵌于待测软件框架中^[5],而逻辑上属于新的 GUI 自动化测试框架。如图 1 所示。

通常将新开发的这类优化软件称作测试马甲 Test Harness,它可以使用待测软件的部分内部函数,并且提供外部接口供其他应用使用,形成一个内外数据交换的通道。区别于传统的 GUI 测试方法,数据链低级

需求测试用例同样可以通过自动化工具完成,增加了自动化测试的用例覆盖率,内部接口的使用,也增强了软件的调试功能,使得 bug 查找信息更加丰富^[6]。

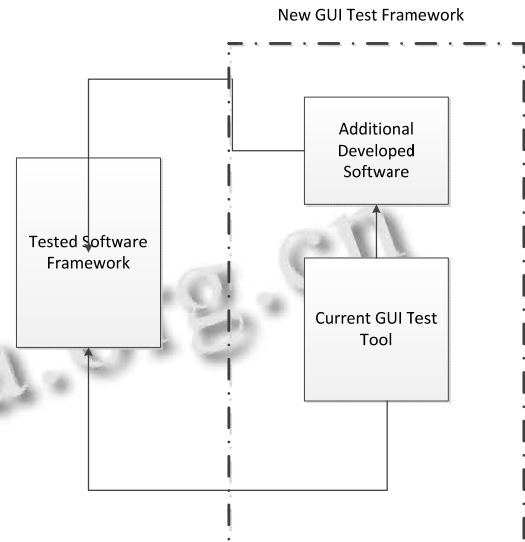


图 1 一种新的 GUI 测试框架

2.2 Test Harness 测试方法

根据软件架构的设计,协议栈中 ATS 和 CSA 的公共协议为 ARINC 619, ARINC 429 和 AFDX; ATS 专有协议为 ARINC 623 和 ARINC 622; CSA 专有的协议为 ASN.1 和 Contractor Manager。具体如图 2 所示。

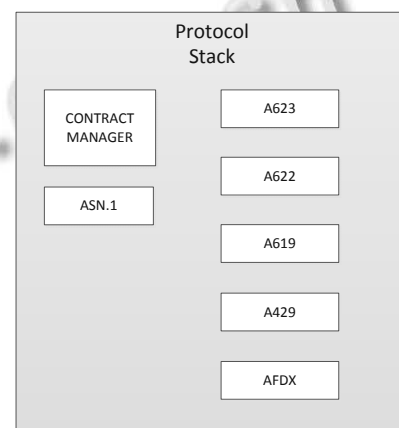


图 2 软件协议栈部署

通过软件最终的 HMI 界面,以及 CMF 和 Ground 模拟器,都只能得到软件级别的最终测试结果,然而我们的需求是按照 block(上图中的一个小单元), module(类似协议栈这样由多个 block 组成的单元)和软件应用级别来进行编写的。因此,基于需求的测试用

例,也相应分为这三个级别.具体如图3.

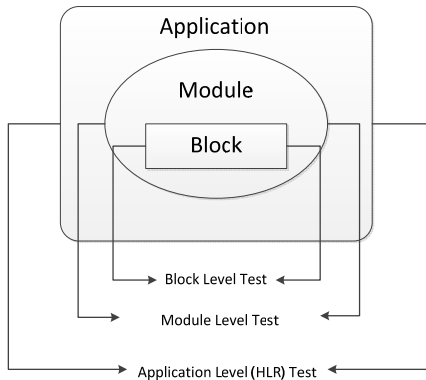


图3 软件测试级别分类

为了得到软件内部各个 block 和 module 的输出结果,同时可以对其进行输入操作,我们采用了 Test Harness 的测试方式. Test Harness 主要用于支持自动化测试,他通过调用 ATS 和 CSA 软件相应的函数,将 block 和 module 的接口提供给测试工具,并且应用于 HMI 界面,进行全面的 GUI 测试.另外,通过 Test Harness,测试工具可以输入越界,非法值等任意参数,对软件进行健壮性测试,然而这样的测试仅仅基于软件本身的 HMI 界面,是无法完成的.

ATS, CSA 和 Test Harness 都将作为独立的软件在 CDS(Cockpit Display System)模拟器上进行显示,软件测试工具可以通过操作 CDS 模拟器的各个界面,对各个 block, module 和 application 的接口输入参数,同时也可以观察到各个输出结果.应用了 GUI 自动化测试之后,由 GUI 自动化测试工具进行统一调度,操作和结果比对,得到测试结果.

Test Harness 必须做到即插即用,插入这段代码和删除这段代码,对 ATS 和 CSA 的软件功能不会造成任何影响.当然,由于这两款软件中 ATS 属于 C 级软件,根据 DO-178C 的要求,我们使用 Test Harness 对 ATS 软件进行测试之前,首先要对 Test Harness 进行 Tool Qualification Verification.同样,GUI 自动化测试工具也必须使得工具认证测试自动化.

2.3 GUI 自动化测试环境搭建及数据流介绍

搭建测试环境,要满足(1)易于测试操作(2)易于问题排查(3)易于模拟器软件部署(4)易于自动化测试实施^[7].因此,在数据链项目中采用了网络直连的方式进行环境搭建,具体如图4所示.

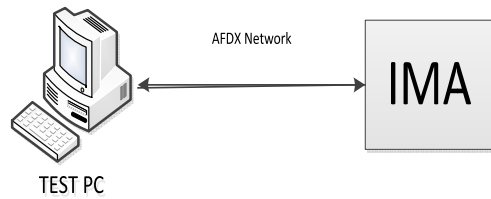


图4 测试环境简图

测试 PC 和 IMA 设备通过 AFDX 网络直连.在 IMA 上,驻留有 CSA 和 ATS 软件,以及 Test Harness 测试工具.在测试 PC 上,部署了以下实体:

- (1)AFDX Driver, 用于管理 AFDX 的 Virtual Link 和封装,解析 AFDX 报文.
- (2)ABLE, 由 Rockwell Collins 提供,可以模拟地面设备和 CMF, Printer, Recorder 等设备的模拟器软件.
- (3)Parameter Simulator 是一个参数模拟器,主要用于广播字的接收显示和周期性数据的发送.
- (4)CDS Simulator 是 HMI 产品的模拟器,用于人机交互,ATS, CSA 和 Test Harness 的操作界面在此显示.
- (5)GUI 自动化测试工具,用于操作测试 PC 上所有的模拟器软件.

测试 PC 的具体部署如图5.

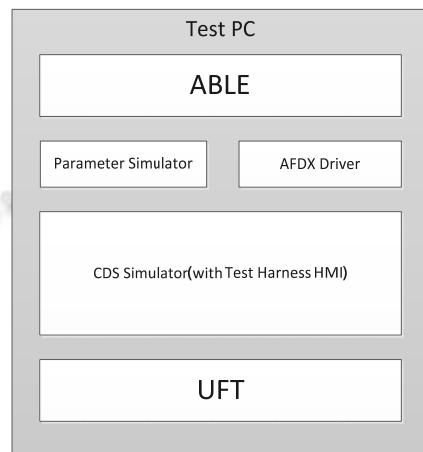


图5 测试机上的软件部署

在此环境中,数据的传输将一目了然.飞机到地面的数据,称为下行数据.

在 CDS 模拟器上通过 ATS 软件的 HMI 界面进行下行数据的发送操作.

这个发送命令通过 AFDX Driver 发送到相应的虚链路当中,再通过测试 PC 上的 AFDX 网卡,发送至

IMA.

IMA 接收到数据后, 进行下行报文的封装处理, 选择相应的虚链路后, 通过 IMA 上的 AFDX 接口发送至测试 PC.

测试 PC 接收到数据后, AFDX 网卡将数据交付给 AFDX Driver, 他按照相应的虚链路信息将数据转发至相应的软件, 此处应该传输给 ABLE.

ABLE 接收到数据后, 对此报文进行回复, 产生新的上行数据发送至 Driver.

上行数据到达 AFDX Driver, 在转交至 AFDX 网卡发送至 IMA.

IMA 接收到这个上行消息后进行处理, 如果需要显示, 将通过 AFDX 网络发送至测试 PC, 在 CDS 模拟器上进行显示.

可以通过图 6 了解这个数据流传输举例.

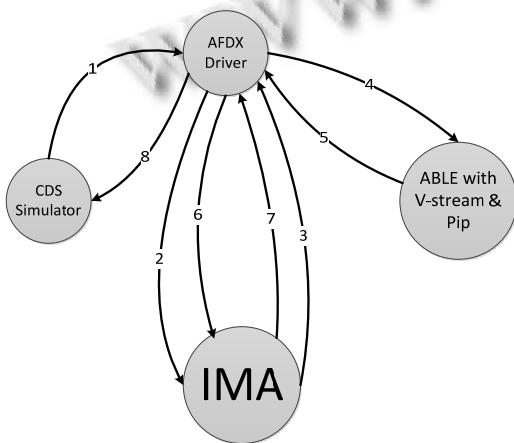


图 6 Application 级别的数据流

Application 级别的测试数据流可以如上图所示, GUI 自动化测试工具只需要操作测试 PC 上的各个软件, 并在测试 PC 的各个软件上获取测试结果. 关于 Module 和 Block 级别的测试, 需要用到 Test Harness. 由于 Test Harness 的操作界面在 CDS 模拟器上, 软件驻留在 IMA 上, 测试时数据流的传输情况, 可以通过测试 ARINC 623 协议举例.

测试 ARINC 623 block 时, 在 CDS 模拟器上通过 Test Harness 的 HMI 界面输入一个下行数据. 这个下行数据发送命令通过 AFDX Driver 发送到相应虚链路中, 发送至 IMA.

IMA 接收后, Test Harness 组装好这个输入报文, 发送至 ARINC 623 block.

经过 block 处理后, 发送数据到 ARINC 622 block, 此刻 Test Harness 截取这个输出报文.

再通过 IMA 上的 AFDX 接口发送至测试 PC.

测试 PC 的 AFDX Driver 将发送至 CDS 模拟器进行显示. 具体数据流见图 7.

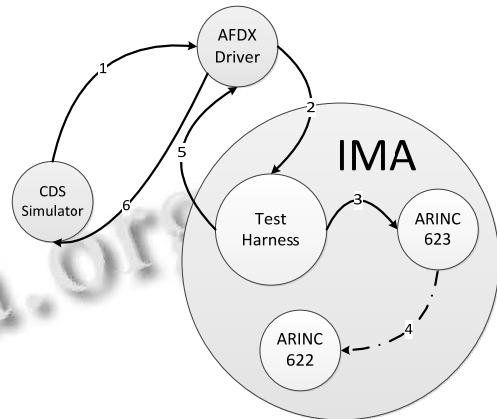


图 7 使用 Test Harness 的数据流

Module 和 Block 级别的测试数据流可以如上图所示, GUI 自动化测试工具也只需要操作测试 PC 上的模拟器软件, 并在测试 PC 上的各个软件上获取测试结果.

2.4 自动化测试结果记录

根据 DO-178C 的要求, 属于 C 级的 ATS 和 D 级的 CSA 两款软件都需要完成基于需求的测试覆盖, 相应的测试结果数据也需要记录. 数据链项目特殊的一点是, CDS 模拟器使用量模拟光标. 当 windows 的光标移至 CDS 模拟器界面时, 光标将自动转换成 CDS 自有的光标, windows 光标的属性将不复存在. 因此, CDS 模拟器上的光标没有复制和粘贴等功能, 只有选择的属性, 测试结果的记录和测试数据的保存必须另辟蹊径.

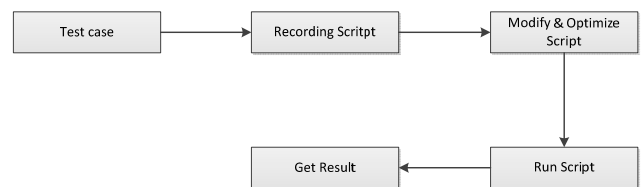


图 8 自动化测试脚本开发流程

如上图所示, 整个自动化测试脚本的开发流程是从测试用例开发完成开始, 根据测试用例人工操作一次, 此刻 GUI 自动化测试工具会自动将测试步骤录制

下来,转换成脚本.初步得到的脚本需要进行部分的修改和优化,并通过多次调试后,得到最终的自动化测试脚本.测试脚本编写完成,整合入库后,就可以通过 GUI 自动化测试工具建立任务,批量运行脚本,最终得到测试结果^[8].

在此期间,多数模拟器软件可以编写脚本,将结果保存为文字文件.而特殊的 CDS 模拟器软件上的测试结果,需要通过编写脚本,对整个 CDS 模拟器界面进行截图,并保存此图片.同样验证点也只能插入在此图片上,将图片作为识别对象进行数据对比,得到测试结果,这就是上文提到的“蹊径”.

因此,通过 GUI 自动化测试工具得到的测试数据和测试结果,不限于形式,图片和文字文件都是可以保存和结果对比的,这些数据也都是可以作为适航证据提供给局方.

2.5 自动化测试的实施

对象库 object repository 的建立是使用 UFT 进行 GUI 测试的最关键步骤,规范和统一对象库,是实施 GUI 自动化测试的最大前提.根据不同的模拟器软件,建立不同的对象库文件,便于管理,也便于将来对对象库的复用和完善.受限于模拟器软件的开发限制,不是所有的对象都可以被 UFT 的 SPY 工具获取,必要时可以使用描述性语言进行脚本的编写,甚至可以考虑插入对象的方式对识别对象进行操作^[9].对象库的初步建立如图 9 所示.

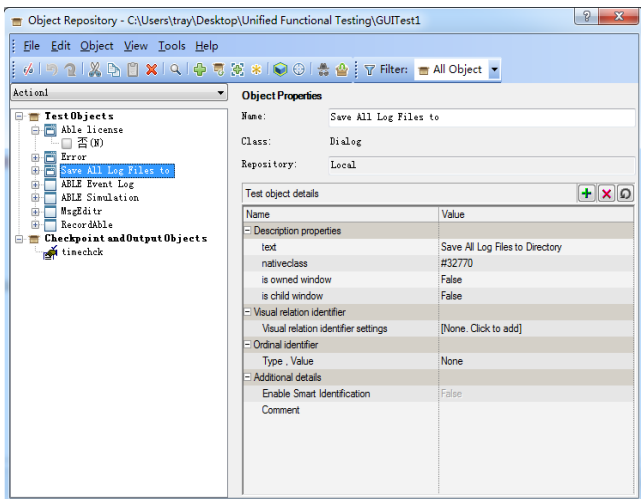


图 9 初步建立的对象库

在初步建立对象库之后,建立过程函数,将部分动作和可重复性操作函数化,建立私有函数库,在

GUI 自动化测试过程中,不断完善私有函数库,将有助于增加自动化脚本运行的稳定性和一致性.如图 10 所示,使用 VBS 编写的简单过程函数,将不同的函数文件关联至新的测试脚本中,便可以轻松调用文件中所有的函数^[10].

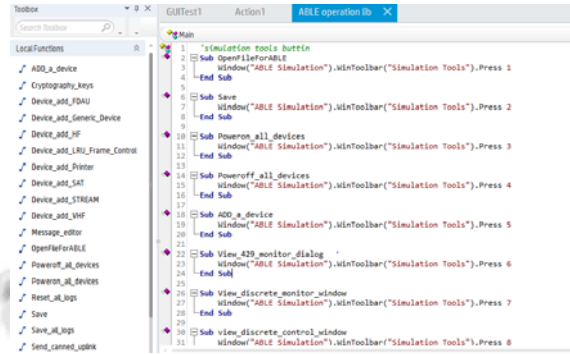


图 10 过程函数库实例

UFT 带有 record 脚本的功能,将测试步骤 record 下来,自动生成测试脚本.在参考多本 GUI 自动化测试书籍和本人实践后,建议不使用该方法,一是录制脚本产生的对象库不规范,存在多余对象,另一个是生成的脚本不规范,需要人工调整.使用 VBS 人工编写脚本,规范脚本编写,调用已有函数库,简化脚本内容,将有助于增加自动化测试脚本运行的可靠性和稳定性.

编写的自动化测试脚本中,根据测试需要,将过程文件进行规范的保存以便于民航局进行适航检查时的证据.UFT 具有自带的 window capture 函数,通过调用此函数,将以 window 为单位的图片数据进行截屏和保存.如图 11 和图 12,分别为 CDS 模拟器软件截图和 ABLE 日志截图.

当然,不同的测试用例有不同的测试验证点,可以通过截图进行对比,保存测试结果和测试证据,也可以通过插入文本验证点,对特殊区域的文本数据进行对比,保存测试结果和测试证据,甚至还可以在 UFT 所截图片中插入文本验证点.灵活使用不同类型的 UFT 自带的 checkpoint 和 regular expression,可以简化脚本和增加验证点比对的正确性^[11].

自动化测试的实施必须因地制宜,充分复用普遍性方式,敢于创新特殊情况下的实施方法,这样才能充分利用个性化的 GUI 自动化测试框架和强大 GUI 自动化测试工具完成最高效的自动化测试.



图 11 CDS 模拟器截图

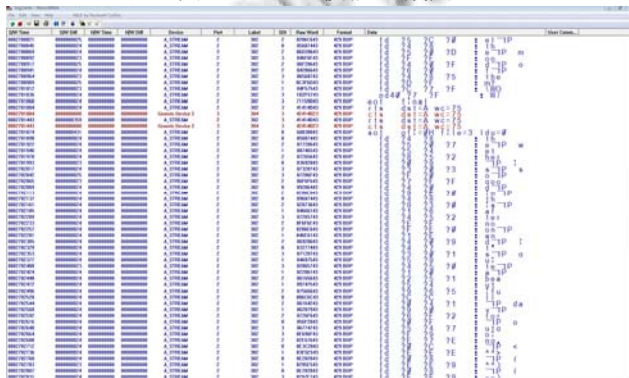


图 12 ABLE 日志截图

3 结语

GUI 自动化测试工具运用在数据链系统中, 打破了民用飞机机载软件的常规测试方案, 也与其他行业的真正的 UI 测试有所不同. 特别是将 Test Harness 测试工具引入到 GUI 自动化测试方案中, 将软件内部模块测试也界面化, 并与 GUI 自动化测试完美结合, 是数据链系统集成自动化测试的一大创新点. GUI 自动化测试工具的引入, 解放了传统的手工测试所需的重复劳作人力, 让测试者更加关注于软件测试的规范性, 创新性和探索性, 更多的去思考如何使得软件更加稳

定的方法. 同时, 也解放了项目中紧张的验证测试环境和开发调试环境, 正常工作时间将环境提供给开发调试和问题排查, 非工作时间在同样环境上运行自动化测试脚本, 自动得到测试结果. 相信在未来的民用飞机机载软件的测试方案中, GUI 自动化测试工具的运用会越来越广泛, 越来越成熟, 全面实施自动化测试必定是民用机载软件测试方案的一大趋势^[12].

参考文献

- 1 刘天华.民用飞机数据链通信管理技术[硕士学位论文].成都:中电科航空电子有限公司,2010.
- 2 余杰,赵旭斌,精通 QTP-自动化测试技术领航.北京:人民邮电出版社,2014:10-89.
- 3 Software Considerations in Airborne Systems and Equipment Certification. Document RTCA/DO-178C, RTCA. Cth ed, 2011: 250-380.
- 4 骆光明.数据链:信息系统连接武器系统的捷径.北京:国防工业出版社,2008:60-61.
- 5 尹亚兰.战术数据链技术及在联合作战中的运用.北京:国防工业出版社,2014:31-33.
- 6 赵志勇,毛忠阳,数据链系统与技术.北京:电子工业出版社,2014:191-195.
- 7 Tarun L.QTP 自动化测试权威指南.北京:人民邮电出版社,2014:53-56.
- 8 Li KL, Wu MQ.图形用户界面测试自动化.北京:电子工业出版社,2005:53-56.
- 9 杨莉,杨合生,软件自动化测试:入门、进阶与实践.北京:电子工业出版社,2012,32(22):61-65.
- 10 Tarun L, Manika G, Chhanda B. Uft/Qtp Interview Unplugged: And I Thought I Knew Uft! USA: Knowledgeinbox, 2013: 101-102.
- 11 田艳琴.QTP 从实践到精通.北京:电子工业出版社,2010, 64-65.
- 12 胡铮.软件自动化测试工具实用技术.北京:科学出版社, 2011,11:85-88.