

文章编号: 1005-8451 (2016) 11-0041-05

基于关键字驱动的自动化测试技术在大数据平台上的应用

杨涛存, 徐贵红, 郭剑峰, 东春昭

(中国铁道科学研究院 铁路大数据研究与应用创新中心, 北京 100081)

摘要: 随着互联网业飞速发展及“大数据”概念的提出, 为提高大量B/S架构应用的测试效率, 本文利用关键字驱动测试技术, 设计开发了一款基于关键字测试脚本的B/S架构应用自动化测试工具——EasyWT, 实现了对Web页面的控件抽取、录制、激励、验证等功能, 基本实现了一个标准化的测试流程。并使用EasyWT在铁路大数据平台上进行了实践应用, 结果表明, 系统能够完整地自动化测试的全部工作; 同时针对实践过程中遇到的一些问题, 提出了一些改进及优化方案。

关键词: 关键字驱动; 测试; 大数据; 软件工程

中图分类号: TP39

文献标识码: A

Keyword-driven based automated test technology applied to big data platform

YANG Taocun, XU Guihong, GUO Jianfeng, DONG Chunzhao

(Research and Application Innovation Center for Big Data Technology in Railway, China Academy of Railway Sciences, Beijing 100081, China)

Abstract: With the rapid development of the Internet industry, and the proposed concept of big data, in order to improve the test efficiency for the application of a large number of B/S architecture, this article designed and developed an automated testing tool of keyword-driven test script based B/S architecture application that was EasyWT. It was implemented the function of control extraction, recording, motivation, verification and etc. on the Web page, implemented a standardized test procedures basically. The application of EasyWT on railway big data platform was carried out. The application result showed that the System could complete all the work of automated testing. Meanwhile, in view of some problems in practice, the article proposed some improvement and optimization schemes.

Key words: keyword-driven; test; big data; software engineering

随着近期“大数据”概念的持续火热, 软件功能需求的日益复杂和软件规模的日趋庞大, 软件产品的质量成为软件生存和发展的关键, 软件测试技术的重要性也更显突出。同时随着互联网和大数据的迅猛崛起, B/S架构应用已经渗透到国计民生的各个领域, 一个非常关键的问题是如何保证B/S架构应用的可靠性和质量。测试是提高软件可靠性和保证软件质量的一种最基本的手段。目前, B/S架构应用的测试主要依赖测试工程师的直觉和经验, 没有比较系统的方法和工具, 测试是一个耗时的、代价昂贵的过程。许多B/S架构应用在没有进行充分测试的情况下投入运营, 质量难以保证, 导致B/S架构应用软件开发可靠性不高、风险大。因此, 迫

切需要实现B/S架构应用测试的新方法和新尝试。

本文的核心实现内容EasyWT (Easy Web Test) 是基于关键字测试的B/S架构应用测试系统, 从B/S架构应用测试的标准流程和软件测试功能要求出发, 设计了一个综合的软件测试集成开发环境, 同时也是B/S架构自动化测试工具, 该工具的主要功能包括Web文档结构分析、录制Web页面动作、回放Web页面动作、关键字测试脚本生成与转换等。

1 需求与原则

EasyWT系统需要的功能, 是按照实际的B/S架构软件测试用例设计的, 它在系统需求分析阶段应该满足以下要求:

(1) 能够独立地访问网站或者依托于网络浏览器;

收稿日期: 2016-05-08

作者简介: 杨涛存, 研究实习员; 徐贵红, 副研究员。

(2) 能够方便地查看 Web 页面源代码并且对关键内容进行标注 ;

(3) 能够解析 DOM 文档, 使之成为测试所需要的数据结构 ;

(4) 提供文本编辑环境, 能够编写测试用例脚本 ;

(5) 能够录制用户动作, 以简化用例脚本的输入 ;

(6) 能够根据用户输入的脚本激励 Web 页面做出相应动作 ;

(7) 能够输出脚本, 并且根据需求, 转换为其他高级语言脚本。

由于 EasyWT 的实现基于现有的开源项目, 在原项目上重新扩展开发, 隶属于整个项目的一部分, 因此, 在进行系统设计的过程中需要遵从最少改动原则、最大复用原则及模块化原则。

2 系统设计

2.1 总体结构

从模块结构上看, 系统总体可分为 4 个模块, 如图 1 所示。

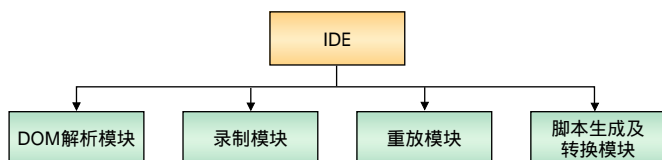


图1 集成开发环境总体模块架构

(1) DOM 解析模块 (DOM Parser) : 对获取的 HTML 的 DOM 结构进行解析, 分析 DOM 结构树的层次和分支关系, 过滤无用的脚本、注释, 筛选出测试常用的关键标签。对所过滤出的标签的属性、方法进行统计和列举, 给出标签可以拥有的属性列表和标签可以响应的事件列表以供用户使用 ; 根据用户需求, 对所分析出的标签按照所需格式入库。

(2) 录制模块 (Recorder) : 是脚本编写的重要辅助工具, 通过对浏览器动作行为的录制, 可以让用户摆脱冗杂的代码输入工作, 进而使用相对直观的操作方式, 而其操作会被自动录制下来, 并且形成相应的代码。

(3) 重放模块 (Web Stimulator) : 也叫 Web 激

励模块, 该模块是测试程序的核心, 它实现了自动化测试中所谓“自动”的内容, 即通过用户输入好的脚本, 按照脚本所给出的操作对象和操作内容, 模拟出就像真有用户操作一样的浏览器动作, 对 Web 页面施以激励, 使之发生相应的改变, 并送出 http 请求, 接收服务器的响应。

(4) 脚本生成及转换模块 (Script Generator) : 是程序可扩展性的保障。将本系统中内部使用的脚本描述语言转换成主流的高级语言脚本, 使得用本程序所编写的脚本也可以应用于其他软件中, 提高了程序的适用范围。

2.2 核心模块

2.2.1 Web对象库构建模块结构设计

根据系统总体设计, DOM 解析模块需对获取的 HTML 的 DOM 结构进行解析, 分析 DOM 结构树的层次和分支关系, 过滤无用的脚本、注释, 筛选出测试常用的关键标签等。

利用 DOM, 开发人员可以动态地创建 XML、遍历文档, 增加、删除、修改文档内容, 具有较好的导航能力。因此, 基于 DOM 解析的应用程序的编写比较简单。然而, 单纯 DOM 解析也不足以满足要求, 根据具体情况, 结合高级语言文本分析的方法, 可以解决这些小问题。

2.2.2 录制模块设计

由于录制模块是脚本编写的重要辅助工具, 通过对浏览器动作行为的录制, 可以简化软件测试人员的代码输入工作, 而且操作相对直观, 简单易用, 只要按照测试说明, 在浏览器中相应操作即可, 而其过程会被自动录制下来, 并且自动生成相应的代码, 从根本上减少了测试人员的编码工作, 也降低了对软件测试人员专业水平的要求, 进而降低了成本。

一般的测试工具录制脚本, 大多数都是通过拦截浏览器收发的 http 请求来实现的。但这种方法事实上并没有办法严谨录制用户对 html 页面的所有操作。根据 EasyWT 系统的需求, 这种录制方式显然难以满足。我们需要通过监听用户对 html 页面的操作来录制脚本, 完全了解用户操作了什么以及 html 页面发生了什么变化, 从而实现所需要的录制。

2.2.3 关键字脚本开发环境模块设计

关键字脚本开发环境主要分为两个子模块：脚本开发环境模块和脚本生成及转换模块。

对于脚本开发环境模块，需要为用户提供脚本编辑环境，满足交互性、易用性等需求即可。而对于脚本生成及转换模块，它是程序可扩展性的保障。本系统中使用的脚本描述语言是一种内部语言，它以 XML 形式承载，结构化存储。

2.2.4 测试脚本执行模块设计

回放模块也叫 Web 激励模块，是测试程序的核心，它通过用户输入好的脚本或者是录制的内容，按照脚本中所给出的操作对象和操作内容，模拟出就像真有用户操作一样的浏览器动作，对 Web 页面施以激励，使之发生相应的改变，并送出 HTTP 请求，接收服务器的响应，实现自动化测试中所谓“自动”的内容。

脚本解析部分负责将操作对象、操作名称对应数据等从脚本中解析出来，由于本程序中脚本选用 XML 形式，又采取 DOM 形式分析 XML，这使得脚本解析部分变得相当容易，仅需调用 DOM 库中的控件解析即可。

事件重现部分采用异步脚本驱动网页的形式，即采用 Javascript 语言对网页进行相应的修改，以模拟用户的行为；由于事件本身包含间隔事件（如 WaitForPage 事件等）且不考虑录制时各事件发生的时间间隔，要回放的事件被送入事件回放队列后，只需紧凑执行即可。

3 系统实现

3.1 对象库数据结构

根据系统对 DOM 解析的要求，需要将 DOM 文档解析为如图 2 所示规则。

标签名[>[attr:属性名[&属性值]][<[attr:属性名[&属性值]]].....]

图2 DOM解析数据结构需求

通过对文档的解析，得出 HTML 标签可以响应的事件主要包括一般事件、页面相关事件、表单相关事件、编辑事件、滚动字幕事件和数据绑定事件等。标签、属性和事件均解析入库后，可实现方便的查询与选择操作。

3.2 脚本开发环境

3.2.1 脚本编辑模块

对于脚本编辑模块，采用了 Firefox 插件所使用的 XUL 语言编写了编辑界面，可以实现对脚本的手工编写，XUL 是为了支持 Mozilla 系列的应用程序（如 Mozilla Firefox 和 Mozilla Thunderbird）而开发的使用者界面标示语言。它是一种应用 XML 来描述使用者界面的标示语言，虽然 XUL 并不是一种公开的标准，但它重用了许多现有的标准和技术，包括 CSS、JavaScript、DTD 和 RDF 等。所以对于有网络编程和设计比其他使用者界面标示语言相对简单。

3.2.2 脚本转换模块

本系统中对脚本的转换模块进行了简单化处理，即认为不同语言间的脚本有着——对应的关系，这样就简化了脚本转换的方法。

以同一个测试用例脚本为例，如图 3 所示。

```
<instruction>
  <object>EasyWT</object>
  <action>open</action>
  <id>/id>
  <data type="text">/test/login.htm</data>
</instruction>
<instruction>
  <object>EasyWT</object>
  <action>type</action>
  <id>name=email</id>
  <data type="text">test@mail.com</data>
</instruction>
<instruction>
  <object>EasyWT</object>
  <action>type</action>
  <id>name=password</id>
  <data type="text">12345</data>
</instruction>
<instruction>
  <object>EasyWT</object>
  <action>click</action>
  <id>css=button.login-submit</id>
  <data>/data>
</instruction>
<instruction>
  <object>EasyWT</object>
  <action>waitForPageLoad</action>
  <id>/id>
  <data type="text">30000</data>
</instruction>
```

```
@Test
public void test1() throws Exception {
    easywt.open("/test/login.htm");
    easywt.type("name=email", "test@mail.com");
    easywt.type("name=password", "12345");
    easywt.click("css=button.login-submit");
    easywt.waitForPageLoad("30000");
}

def test_tl(self):
    sel = self.easywt
    sel.open("/test/login.htm")
    sel.type("name=email", "test@mail.com")
    sel.type("name=password", "12345")
    sel.click("css=button.login-submit")
    sel.wait_for_page_to_load("30000")
```

图3 同一脚本的不同语言形式

图 3 中，左侧是 XML 形式原脚本，右上方是 Java 脚本，右下方是 Python，可以很明显看出，除了一些语法性的结构如头部（header）和尾部（footer）以外，脚本部分内容是一一对应的。

3.3 录制模块

3.3.1 监听器

EasyWT 中使用的 Web 事件监听器，所谓监听就是对 DOM 解析出的全部事件做出响应，这样就达到了监听的目的。

监听器找到 DOM 中所有可点击的标签元素，找

到后,对他们进行所有事件的监听,即绑定事件响应。

3.3.2 定位元素

对于发生事件元素的定位,最初采用的是自定义方式,首先根据标签的id属性,如果没有id属性,就根据其他属性的组合来唯一确定标签。系统采用XPath定位的方式来解决这个问题。XPath是一门在XML文档中查找信息的语言,用于在XML文档中通过元素和属性进行导航。它使用路径表达式进行导航,这种表达式也是一个W3C标准。

3.3.3 记录动作

监听器监听到事件后,会将其记录到全局变量的log当中,其中log类的成员变量instruction按照前文约定的XML形式脚本的结构,将此变量进行相应的输出。

3.4 执行模块

3.4.1 解析命令

由于使用了XML形式的内部脚本和DOM解析器,这使得解析命令的工作变得异常简单,它并不涉及编译原理中的语法及词法分析,因为通过DOM解析器已经可以把命令解析为结构化的数据并存在可执行代码中。

3.4.2 执行激励

对于Web页面的激励,最好的方式莫过于用脚本语言,而诸多脚本语言中,Javascript由于其得天独厚的优势,无疑是做Web激励的首选。

执行激励先要获取命令,将命令的对象、动作和数据发送给命令执行模块。所有指令都有对应的输出脚本,对于生成的代码,Javascript提供了动态执行函数eval,只需调用该函数便可执行该代码。

4 EasyWT在大数据平台上的应用

在完成系统的所有重要模块后,将模块进行集成,得到完整的基于关键字脚本的应用测试系统EasyWT。经过调整一些细节,消除代码集成中的错误之后,对系统进行整体功能测试。

整体功能测试的目标是验证系统的功能符合系统设计的目标,整个系统正常工作,在实验中验证结果的正确性。

某铁路局检测车应用系统是一个基于B/S架构

的铁路大数据应用系统。使用EasyWT在该系统上进行测试,以验证EasyWT在大数据系统上的有效性,其界面如图4所示。

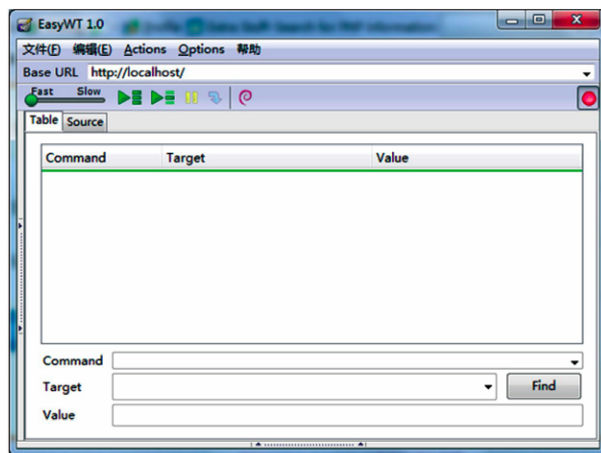


图4 主界面简介

用浏览器打开被测网站,然后打开EasyWT。本次为测试准备的被测网站是一个用户登录页面,如图5所示。



图5 被测页面

根据测试用例,我们在网页上进行简单的操作,注意,此时EasyWT的录制开关是自动打开的,也就是说,这个时候就已经开始了录制动作;如果想对录制下来的脚本进行修改,可以通过点击要修改项,然后在下面的下拉框中直接修改,或者直接点击Source选项卡,修改代码,如图6所示。

代码修改完成后,如果想执行此测试用例,即使用回放功能,需先点击录制按钮暂停录制,然后从

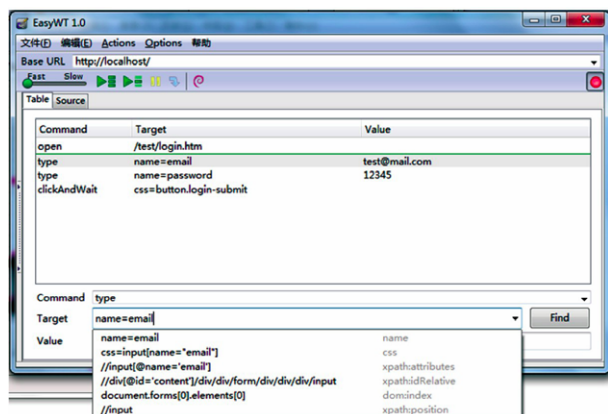


图6 编辑脚本

工具栏中点击执行测试用例按钮，为了达到更直观的演示效果，可以在测试用例中增加一些断点，点击暂停恢复按钮可以一步一步地执行测试用例，否则程序会一次执行完整个测试用例。

5 结束语

测试是软件开发过程中一个不可或缺的部分，随着软件工程学的不断发展，软件测试在整个开发过程中的作用凸显，而在大数据应用平台上各种有关软件测试的新方法、新工具层出不穷，关键字驱动测试技术就是一种新的测试方法，而Web应用测试又是软件测试的一个热点领域，本文中，对这两方面的知识进行了融合，开发了基于关键字驱动脚本的Web应用测试系统—EasyWT，该系统采用了主流的录制回放模式，同时提供了脚本转换功能，使得本系统中的测试脚本能够在其他测试工具中运行。

目前，系统现有功能已经可以成为一个相对完整的逻辑体，但是，对比于QTP等商业软件，以及作为大数据应用测试的工具，还需要在功能和结构等方面进一步完善。

参考文献:

- [1] JC Laprie. Dependable computing and fault tolerance: concepts and terminology[C]. Fault-Tolerant Computing, ' Highlights from Twenty-Five Years ', 2005: 2-11.
- [2] 王卫东, 徐贵红, 刘金朝, 等. 铁路基础设施大数据的应用与发展[J]. 中国铁路, 2015(5): 1-6.
- [3] 缪淮扣, 陈圣波, 曾红卫. 基于模型的Web应用测试[J]. 计算机学报, 2011, 34(6): 1013.

- [4] Rashmi, Neha Bajpai. A Keyword Driven Framework for Testing Web Applications[J]. International Journal of Advanced Computer Science and Applications, 2012, 3(3): 8-9.
- [5] ISO/IEC 9126, Software engineering—Product quality is an international standard for the evaluation of software quality[S]. ISO/IEC:1991.
- [6] 教育部考试中心. 全国计算机等级考试四级教程—软件测试工程师[M]. 北京: 高等教育出版社, 2010: 9-10.
- [7] Fevzi Belli, Mutlu Beyazit, Atif Memon. Testing is an Event-Centric Activity[A]. Proceedings of the International Conference on Software Security and Reliability[C]. 2012:1.
- [8] Pekka Laukkanen. Data-Driven and Keyword-Driven Test Automation Frameworks[D]. Finland: Helsinki University of Technology, 2006:27.
- [9] 冯玉才, 唐艳, 周淳. 关键字驱动自动化测试的原理和实现[J]. 计算机应用, 2004(8): 140-141.
- [10] 陈能技. QTP 自动化测试进阶[M]. 北京: 电子工业出版社, 2010: 15-20.
- [11] 朱宏峰. 基于关键字驱动的web自动化测试的研究[J]. 硅谷, 2009(24): 51.
- [12] Cyntica Eaton, Atif Memon. Evaluating Web Page Reliability across Varied Browsing Environments[A]. US: University of Maryland, 2004:1.
- [13] 胡圣厦. 基于XML的GUI录制-回放技术研究及实现[D]. 成都: 四川大学, 2005.

责任编辑 付思

