

基于数据驱动的软件自动化测试框架

朱 菊, 王志坚, 杨 雪

(河海大学, 江苏 南京 210098)

摘 要: 为了提高软件测试自动化程序的可复用性, 以及在系统、产品以及数据方面的独立性, 在对几种基本的软件测试框架进行研究的基础上, 结合在实际的项目中的软件自动化测试的实践, 提出一种基于数据驱动的自动化测试框架(TAF)。分析了软件测试以及软件测试自动化的现状, 阐述了软件测试自动化框架的策略目标、关键的成功因素, 以及5种基本的软件自动化测试框架。最后提出新的软件自动化测试框架。

关键词: 软件自动化测试; 测试用例; 测试组合; 数据驱动

中图分类号: TP311.56

文献标识码: A

文章编号: 1673-629X(2006)05-0068-03

A Software Automation Test Frameworks Based on Data - Driven Automation Methodology

ZHU Ju, WANG Zhi-jian, YANG Xue

(Hohai University, Nanjing 210098, China)

Abstract: In this paper, describe the strategy targets of software automation test framework. Also, analyze the critical factors of successfully integrating the automation test framework into a project. After a comparative study over five existing products, propose a new software automation test framework, which is called TAF. This framework is highly independent from software systems, products and data. Moreover, it supports data - driven automation methodology. It is integrated into real life project and achieve a good performance.

Key words: software automation testing; test case; test suite; data - driven

0 引言

在一个项目中, 究竟有没有时间去测试所有软件? 答案一定是“NO”。总是有更多的测试需要去做, 或者是测试另一个新的平台, 或者是想去尝试不同的配置^[1,2]。但是随着交付时间的逼近, 分配给每个测试周期的时间变少。那么软件测试项目的经理以及测试小组的成员怎样处理这样的问题呢? 典型的做法是减少每个为产品交付做准备的测试周期的测试数量。这样的经历你有过吗? 当然, 理想的做法是, 先做一些基于风险的分析来决定哪些测试是可以去掉的。然而在实际过程中常常是: 测试小组仅仅把整个测试周期集中在验证修正过的缺陷, 甚至没有足够的时间去完成这些简化的测试计划^[3]。

通过使用自动化测试可以缓解这种困境。而且软件自动化测试框架使得测试脚本的维护量减至最少。然而, 各测试工具提供商多年主打的传统的“捕获/回放”模型导致了高的异乎寻常的脚本维护量, 因为测试数据在测试脚本程序中是以硬编码方式实现的。此外, 工具内建的测试

用例除了测试应用程序的图形用户界面, 实际上并没有用处。因此, 如何选择一个合适的测试自动化框架, 是一个自动化测试小组开始启动前需要最优先考虑的一个问题^[4]。

一个自动化测试框架就是一个由假设、概念以及为自动化测试提供支持的实践的集合。文中描述了5种基本的自动测试框架: 模块化测试脚本框架, 测试库构架框架, 关键字驱动/表驱动测试框架, 数据驱动测试框架, 以及混合测试框架。并且基于此提出一个基于数据驱动的方法学的实用自动化测试框架。

1 自动化测试框架的关键成功因素

为了最大限度地使用测试框架, 需要使其可复用、可管理。下面所描述的是开发自己的自动测试框架所需要遵从的一些基本的指导原则^[5]:

- 要重视测试自动化, 一旦决定要自动化, 就必须当成主要的工作来做。

- 测试设计过程和测试自动化框架必须作为两个单独的实体来开发。

- 测试框架应该独立于应用程序。

- 测试框架应该易于扩展、维护和增强。

- 测试策略/设计词汇表应该独立于框架。

收稿日期: 2005-08-24

作者简介: 朱 菊(1980-), 女, 江苏淮安人, 硕士研究生, 研究方向为软件质量管理; 王志坚, 教授, 博士生导师, 研究方向为网络计算机应用技术、软件复用、网络软件系统集成技术。

·测试策略/设计应该对测试者隐藏测试框架的复杂性。

2 基本的软件测试自动化框架

以下论述5种基本的软件测试框架^[6,7]。测试小组可以根据实际需要去考虑采用其中的一种测试框架而不是仅仅依赖于一个简单的捕获工具。同时,这些框架是了解自动测试框架以及根据自己的需要和经验来设计自己的自动测试框架的基础。

2.1 模块化测试框架

模块化测试脚本框架(Test Modularity Framework)需要创建小而独立的可以描述的模块、片断以及待测应用程序的脚本。这些树状结构的小脚本组合起来,就能组成能用于特定的测试用例的脚本。

在这5种框架中,这个应该是最容易掌握和使用的。在一个组件上方建立一个抽象层使其在余下的应用中隐藏起来,这是众所周知的编程技巧。这把应用同组件中的修改隔离开来,提供了程序设计的模块化特性。模块化测试脚本框架使用这一抽象或者封装的原理来提高自动测试组合的可维护性和可升级性。

2.2 测试库框架

测试库框架(Test Library Architecture)与模块化测试脚本框架很类似,并且具有同样的优点。不同的是测试库框架把待测应用程序分解为过程和函数而不是脚本。这个框架需要创建描述模块、片断以及待测应用程序的功能库文件(例如 SQABasic libraries, APIs, DLLs 等)。

2.3 关键字驱动或者表驱动测试框架

对于一个独立于应用的自动化框架,关键字驱动(Keyword-driven)测试和表驱动(Table-driven)测试是可以互换的术语。这个框架需要开发数据表和关键字。这些数据表和关键字独立于执行它们的测试自动化工具并可以用来“驱动”待测应用程序和数据的测试脚本代码。关键字驱动测试看上去与手工测试用例很类似。在一个关键字驱动测试中,把待测应用程序的功能和每个测试的执行步骤一起写到一个表中。

这个测试框架可以通过很少的代码来产生大量的测试用例。同样的代码在用数据表来产生各个测试用例的同时被复用。

2.4 数据驱动测试框架

数据驱动(Data-driven)测试是一个框架。在这里测试的输入和输出数据是从数据文件中读取(数据池、ODBC 源, cvs 文件, excel 文件, DAO 对象, ADO 对象等),并且通过捕获工具生成或者手工生成的代码脚本被载入到变量中。在这个框架中,变量不仅被用来存放输入值还被用来存放输出的验证值。整个程序中,测试脚本来读取数值文件,记载测试状态和信息。这类似于表驱动测试。在表驱动测试中,它的测试用例是包含在数据文件而不是在脚本中,对于数据而言,脚本仅仅是一个“驱动器”,或者

是一个传送机构。然而,数据驱动测试不同于表驱动测试,尽管导航数据并不包含在表结构中。在数据驱动测试中,数据文件中只包含测试数据。

这个框架意图减少你需要执行所有测试用例所需要的总的测试脚本数。数据驱动需要很少的代码来产生大量的测试用例,这与表驱动极其类似。

2.5 混合测试自动化(Hybrid Test Automation)框架

最普遍的执行框架是上面介绍的所有技术的一个结合,取其长处,弥补其不足。这个混合测试框架是由大部分框架随着时间并经过若干项目演化而来的。

3 数据驱动的自动化测试框架(TAF)

根据上面对于测试自动化以及测试自动化框架的重要成功条件以及自动测试框架的策略目标,在对现有的5种测试框架进行研究的基础上,结合在软件测试方面的实践,提出一个数据驱动的自动化测试框架 TAF。在这里首先阐述该自动化测试框架的体系结构。

3.1 即插即用架构

TAF 是基于即插即用的构架,并且是作为一个模块来设计和实现的。它具有以下特点:

- 系统独立性;
- 产品独立性;
- 数据独立性;
- 次序独立性。

也就是说当测试系统、测试数据和测试次序改变时不需要修改代码。数据驱动引擎被设计并实现来支持现有模块的复用。当测试系统、测试数据和测试次序改变时,仅仅需要改变配置文件、测试用例表以及数据文件。当引入新的测试或者新的验证行为时,仅仅需要实现新的模块或者功能。新的模块一旦创建就可以被应用,仅仅需要对数据驱动引擎的头文件做些许的修改,并不需要复杂的设置就可以象其他模块一样使用这些功能。

3.2 TAF 支持的自动化方法学

软件自动化测试框架 TAF 支持数据驱动的测试用例自动方法学。下面简要介绍这种方法学,以及 TAF 如何实现对它的支持。

数据驱动的测试自动化方法学对于一个产品中普通的可重复的测试用例非常有效。它极大地降低了编码的工作量。TAF 通过实现一个简单数据驱动引擎来支持数据驱动的自动化测试。因此仅仅需要以文本形式创建测试场景文件和测试数据就可以来自动化一个测试用例,并不需要额外编码,前提需要的功能模块已经实现。数据驱动引擎通过解析测试用例文件来调用并执行相应的功能函数,以及函数执行过程中所需要的数据文件。其中每个功能函数都是用脚本语言 perl 来实现,并且每个 XML 文件对应一个独立的测试用例^[8]。

3.3 TAF 组件

TAF 由主程序、模块、测试数据、测试组合、配置文

件、支持工具和测试结果等组件构成。

3.3.1 主程序

当运行一个用 TAF 来开发的 Testware 时,主程序 TAFMain.pl 被调用执行。它根据对配置文件 TestConfig.ini 文件的解析结果来确定运行什么测试组合或测试用例,同时触发数据驱动引擎来解析测试用例文件,并根据解析结果来发出相应的功能函数,以及调用相应的数据文件来执行测试。

3.3.2 模块

TAF 中的模块包括框架以及公共模块、专业模块、产品特定的模块。框架和公共模块包含一些框架和公共函数,例如数据驱动引擎。而产品特定的模块包括测试待测产品或者应用所需要调用的功能函数。专业模块则包括处理特定的功能或者协议所需要的支持函数,例如邮件处理中测试中对协议的支持模块。这些功能模块都放在函数库 lib 中。

3.3.3 测试数据

测试数据是特定测试产品和测试组合的测试数据。对于不同产品测试数据是不一样的。譬如对于文件传送功能的测试数据则表现为各种类型的文件。

3.3.4 测试组合

测试组合为一个测试用例的集合,被指定来完成一个特定的测试目标。它可以被设计来测试一个函数、一个模块,或者是执行一个类型的测试,例如 Release Acceptance Test。而测试数据是一个基本的测试单元,它定义测试状态的初始化、测试步骤、应用在每一步中的测试数据以及其预期结果。在 TAF 中,每个测试用例表现为一个独立的 XML 文件。而每个测试组合则对应一组 XML 文件的集合。

3.3.5 Testware 的配置文件

Testware 配置文件记录执行 Testware 的一些基本配置项,包括文件目录、数据目录、测试组合目录、log 目录以及一些服务、proxy 的配置等。

3.3.6 测试结果

TAF 在执行完一个测试后产生三种类型的测试结果、日志文件、报告和相应的测试过程数据。

3.4 测试流程

图 1 描述了自动化测试框架 TAF 的设计。

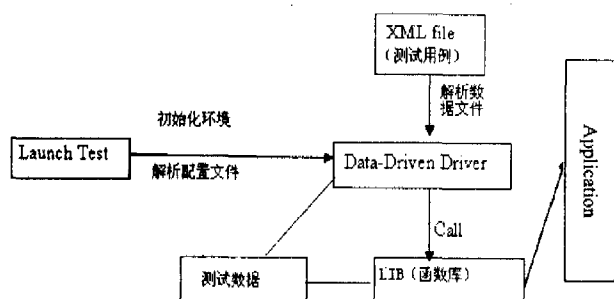


图 1 自动化测试框架 TAF

如同上文所述,这个框架本身由 TAF 主程序、配置文件、数据驱动的 engine、Data(XML file),以及功能函数所定义。

当运行一个使用 TAF 开发的 Testware 时,主测试被执行。TAF 的主程序是一个 perl 脚本程序。它初始化测试环境,解析配置文件,启动数据驱动引擎(Data-driven engine)。

进行测试时数据驱动引擎取出以 XML 文件组织的测试用例,并根据解析的结果调用相应的功能函数并获取相应的测试数据来执行测试。测试结束后生成相应的测试结果。在这里每个测试用例是一个独立的 XML 文件。

整个自动化测试依赖于启动整个测试的主程序,以 XML 文件来组织的测试用例、数据驱动引擎、测试数据以及功能函数库。数据驱动测试引擎根据对数据文件的解析结果,决定要调用的功能函数,然后通过驱动自动测试脚本调用功能函数来执行测试用例。

4 结 论

软件自动化测试框架 TAF 是以数据驱动的方法学为基础来设计的,可以应用在各种不同类型项目的测试过程中。当然 TAF 作为一个自动测试框架,其结构也是开放的、不固定的,使用时可以根据项目自身的特点对其进行扩展和改进。

参考文献:

- [1] Bach J. Test Automation Snake Oil[EB/OL]. <http://www.satisfice.com/articles/test-automation-snake-oil.pdf>, 1999.
- [2] Marick B. When should a test be automated[EB/OL]. <http://www.stickyminds.com/sitewide.asp?ObjectId=2010&Function=DETAILBROWSE&ObjectType=ART>, 1999.
- [3] Mosley D J, Posey B A. Just Enough Software Test Automation[EB/OL]. <http://www.phptr.com/mosley,2002-07>.
- [4] Pettichord B. Success with Test Automation[EB/OL]. <http://www.io.com/wazmo/succpap.htm>, 1996-05.
- [5] Mosley D J, Posey B A. 软件测试自动化 Just Enough Software Test Automation[M]. 邓波, 黄丽娟, 曹青春译. 北京:机械工业出版社, 2003.
- [6] Kelly M. Choosing a test automation framework[EB/OL]. <http://www-128.ibm.com/developerworks/rational/library/591.html#N10223>, 2003-11-20.
- [7] Nagle, Carl. Test Automation Frameworks[EB/OL]. <http://safesdev.sourceforge.net/FRAMESDataDrivenTestAutomationFrameworks.htm>, 2000.
- [8] Young D. Test Automation: An Architected Approach[EB/OL]. <http://www.stickyminds.com/sitewide.asp?ObjectId=8336&Function=DETAILBROWSE&ObjectType=ART>, 2004.