

# 软件测试中的测试用例及复用研究

尚冬娟,郝克刚,葛 玮,李海军

(西北大学 计算机科学系,陕西 西安 710069)

**摘 要:**软件测试过程中产生的大量测试用例对保证软件的质量起到关键作用,对于专门从事软件测试的机构来说,测试用例的复用尤其重要。在测试用例的设计、选择的基础上,构造出基于复用的测试用例,同时采用测试用例库管理的方法来实现测试用例的复用,可以提高软件测试的工作效率。

**关键词:**测试用例;复用;测试用例复用;测试用例库

**中图分类号:**TP311.56

**文献标识码:**A

**文章编号:**1005-3751(2006)01-0069-04

## A Study of Test Cases and Reuse in Software Testing

SHANG Dong-juan, HAO Ke-gang, GE Wei, LI Hai-jun

(Department of Computer Science, Northwest University, Xi'an 710069, China)

**Abstract:** A large number of test cases play a key role in guaranteeing the quality of software during the software testing. To the organization specializing in software test, it is especially important that test cases are reused. Discussed mainly that design and choice test cases, and put forward constructing reuse-based test cases. The ones that realize test cases reuse by testing with management of test case library in order to improve the working efficiency of software testing.

**Key words:** test case; reuse; test case reuse; test case library

### 0 引 言

随着软件产业化发展,软件的功能越来越强大,软件的复杂程度也越来越高。复用技术的发展,大大提高了基于组件的软件开发的效率和质量。软件测试(Software testing)作为软件生存期(Software life cycle)中的一个重要阶段,在软件投入使用前,对需求分析、概要设计、详细设计和编码的最终复审,是软件质量保证的关键步骤,其目的是为了检验软件系统是否满足需求。一个组件测试包的源代码中含有如何使用该组件的大量信息,测试文档则说明了组件的实用性和对用户的技术要求,测试结果还能够提供给用户关于该组件的可靠性的基本信息。更重要的是使用者能够快速有效地使用复用的测试来验证该组件应用的正确性,并最终验证组件在该应用中的可靠性<sup>[1]</sup>。

文中主要是面向第三方的软件测试机构,讨论了在现在的软件测试中,测试用例的设计、选择和执行;并在此基础上,对测试用例实行有效的管理,实现测试用例的复用,最终达到改善软件测试环境,提高软件测试效率的目的。

### 1 软件测试中的测试用例

测试用例(Test case)实际上是对软件运行过程中所有有可能存在的目标、运动、行动、环境和结果的描述,是对客观世界的一种抽象。测试用例体现了一定的测试方案、方法、技术和策略,内容包括测试目标、测试环境、输入数据、测试步骤、预期结果、测试脚本等,并形成文档。

#### 1.1 测试用例在软件测试中的作用

影响软件测试的因素很多,例如软件本身的复杂程度,开发人员(包括分析、设计、编程和测试人员)的素质,测试方法和技术的运用等等。测试用例是测试工作的指导,可以把人为的因素减少到最小,而且会随着测试的进行日趋完善,是软件测试质量稳定的根本保障。

测试用例在软件测试中的作用主要表现在:

- 1) 指导测试的实施;
- 2) 规划测试数据的准备;
- 3) 编写测试脚本的“设计规格说明书”;
- 4) 评估测试结果的度量基准;
- 5) 分析缺陷的标准(需进一步查询参考资料,确认)。

测试用例对测试工作的控制和指导作用相当于设计文档对编码的指导作用,尤其在大系统表现出对系统测试的权威性。

所以说,软件测试的核心任务是生成和执行软件测试用例,以验证软件的质量。而且,往往一个好的软件测试包能够体现软件测试思想、技巧,同时还保存有大量的测

收稿日期:2005-04-04

基金项目:国家“八六三”项目(2004AA115090)

作者简介:尚冬娟(1973—),女,山西运城人,硕士研究生,研究方向为软件工程与理论;郝克刚,教授,博士生导师,研究方向为软件工程与理论。

试数据、结果以及测试过程记录。如果能有效地将这些东西复用,将极大地提高软件测试的效率。

## 1.2 测试用例的设计

测试用例属于软件测试工作的指导性文件,测试用例的优劣直接影响软件测试的质量。如何以最少的人力、资源投入,在最短的时间内完成测试,发现软件系统的缺陷,保证软件的优良品质,是软件公司探索和追求的目标。尤其对第三方测试机构来说,软件测试中的测试用例设计及复用更为重要。

设计测试用例即设计针对特定功能或组合功能的测试方案,并编写成文档。测试用例应该体现软件工程思想和原则。测试用例的选择既要有一般情况,也应有极限情况以及最大和最小的边界值情况。因为测试的目的是暴露应用软件中隐藏的缺陷,所以在设计选取测试用例和数据时要考虑那些易于发现缺陷的测试用例和数据,结合复杂的运行环境,在所有可能的输入条件和输出条件中确定测试数据,来检查应用软件是否都能产生正确的输出。

表 1 测试用例一般描述

|                   |   |             |                                    |
|-------------------|---|-------------|------------------------------------|
| 测试类型:[填写单元/集成/系统] |   | 日期:[填写编写日期] |                                    |
| 测试用例编号            | [必写,填写编号]   | 设计者         | [必写,填写姓名]                          |
| 测试项               | [必写,填写所测试项名称]   | 所属功能        | [必写,测试项的功能]                        |
| 功能描述              | [必写,被测试项的功能描述]  |             |                                    |
| 测试要点              | [必写,规定本模块主要测试内容以及业务逻辑规定(数据的完整性,正确性,与业务的符合性,是否可删除/作废)] |             |                                    |
| 预置条件              | [必写,哪些测试用例必须执行(为了运行该模块,哪些模块应该先启动)]                    |             |                                    |
| 输入数据              | [必写,需要输入系统的测试数据]                                      | 预期结果        | [必写,说明进行该项测试会得到怎样的结果才正确,是实际测试者的参考] |
| 测试说明              | [可选,一些特殊说明,如操作上应该注意的问题等]                              |             |                                    |
| 影响范围              | [必写,指出涉及到什么模块/文件/函数等]                                 |             |                                    |
| 审核人               | [填写意见/姓名]   | 日期          | [填写审核日期]                           |

表 2 测试用例描述案例

|           |  |               |   |
|-----------|--|---------------|---|
| 测试类型:单元测试 |  | 日期:2000/08/10 |   |
| 测试用例编号    | T-03   | 设计者           | 尚东娟   |
| 测试项       | 用户管理   | 所属功能          | 安全管理  |
| 功能描述      | 系统中有用户管理权限的用户可进行如下的用户操作:<br>1. 添加:向用户表中添加一个新的用户;<br>2. 删除:从用户表中删除一个用户;<br>3. 查询:查询用户表中选定的用户类型(类型:系统普通用户,高级用户,超级用户)   |               |   |
| 测试要点      | 1. 用户编号:4 位数字,唯一性<br>2. 用户姓名:10 位字符,是否合法用户<br>3. 用户口令设置:8 位字符<br>4. 增、删、查操作正确性   |               |   |
| 预置条件      | 启动系统/安全管理/用户管理   |               |   |
| 输入数据      | 1. 启动用户管理,按“增加”按钮<br>2. 正确输入用户(sss),编号(0006),口令(user),类型(1)存盘<br>3. 按“查询”按钮<br>4. 关闭查询窗口,按“增加”<br>5. 输入用户(张三)不输入编号,其他信息正确,存盘<br>6. 输入一位编号,移动光标<br>7. 输入姓名中存在非数字/字母符号,移动光标<br>8. 输入用户编号(0006),移动光标<br>... | 预期结果          | 1. 弹出增加用户界面<br>2. 返回用户管理界面<br>3. 在用户列表中查询(sss)的信息<br>4. 弹出增加用户界面<br>5. 提示“缺编号,不能存盘”<br>6. 编号自动补足 4 位<br>7. 提示姓名存在“非法字符”<br>8. 编号重复<br>... |
| 测试说明      | 编号的唯一性,口令加密,操作的方便性   |               |   |
| 影响范围      | 影响的数据库有:YHGLK, YHQAQK  |               |   |
| 审核人       | 李静   | 日期            | 2000/08/20  |

测试用例的一般描述见表 1<sup>[2]</sup>、描述案例见表 2<sup>[2]</sup>。测试用例的描述明确了所测试项的功能的概要描述、测试要点、测试预置条件、测试系统输入数据、预期产生结果、测试说明及影响范围。测试输入数据和预期产生的结果是测试用例的核心内容,测试输入数据应该全面覆盖测试功能。通常测试的数据包括:正确测试数据 40%,边界测试数据 30%,错误测试数据 30%,预期结果是对每个测试输入数据进行定量准确的描述。

## 1.3 测试用例的选择

基于复用的目的,在选择测试用例时,要针对不同的测试方法进行明确的选择。

任何工程化的产品都有两种测试方法,一种方法是已知产品应该有的功能,通过测试检验每个功能是否都能正常使用;另一种是已知产品内部工作过程,通过测试产品内部动作是否按照说明的规定正常进行。即所谓的黑盒测试和白盒测试方法。测试用例的设计将根据两种测试方法的特性定制。

●黑盒测试完全不考虑程序的内部结构和处理过程。设计测试用例旨在测试软件是否满足功能要求,主要诊断下列几类错误:

- ①不正确或遗漏的功能;
- ②界面错误;
- ③数据结构或外部数据访问错误;
- ④性能错误;
- ⑤初始化和中止条件错误。

●与黑盒测试相反,白盒测试密切关注处理细节,根据程序的控制结构设计测试用例,原则是:

- ①保证模块中的每一独立的路径至少执行一次;
- ②保证所有判断的每一分支至少执行一次;
- ③保证每一循环都在边界条件和一般条件下至少各执行一次;
- ④验证所有内部数据结构的有效性<sup>[3]</sup>。

## 2 复用的概念

### 2.1 软件复用

软件复用是将已有的软件及其有效成分用于构造新的软件或系统。其出发点是应用系统的开发不再采用一切“从零开始”的模式,而是以已有的工作为基础,充分利用过去应用系统开发中的经验知识和经验,如需求分析结果、设计方案、源代码、设计计划及测试案例等,从而将开发的重点集中于应用的特有构成成分。

软件复用可以从多个角度进行考察。依据复用的对象,可以将软件复用分为产品复用和过程复用。产品复用指复用已有的构件,通过构建集成得到新系统。过程复用指复用已有的软件开发过程。依据对可复用信息进行复用的方式,可以将软件复用分为:白盒复用和黑盒复用。黑盒复用一般为对已有产品或构件不需作任何修改,直接进行复用。它主要基于二进制代码的复用,包括可执行程

序的复用和基于库(包括动态链接库和静态库)的复用。白盒复用一般为源代码一级的复用,及相应的测试用例和文档的复用。

软件复用有3个基本问题:一是必须要有可以复用的对象;二是所复用的对象必须是有用的,三是复用者需要知道如何去使用被复用的对象。通过软件复用,消除了包括分析、设计、编码、测试等在内的许多重复劳动,从而提高了软件开发的效率,同时,通过复用高质量的已有开发成果,避免了重新开发可能引入的错误,从而提高了软件质量<sup>[4]</sup>。

## 2.2 软件测试复用

基于复用的软件测试是软件开发过程中的一个重要阶段,是“为了发现错误而执行程序的过程”。它是根据软件开发各阶段的规格说明和程序的内部结构而精心设计一批测试用例,并利用这些测试用例去运行程序,以发现错误或缺陷。因此,软件测试的复用主要包括测试过程的复用、测试方法的复用和测试技巧的复用。

(1)测试过程的复用,就是测试流程的复用,采用既定的测试流程,针对被测项目的需求,进行裁减,使测试的流程规范化。

(2)测试方法的复用,主要指测试计划的设计、测试策略的采用、测试项细则的编写、软件缺陷的分析和测试报告的撰写等方面的复用。

(3)测试技巧的复用,主要指测试用例的复用。在特定的情形下,选择何种测试用例是发现软件问题的关键因素。如果将大量的测试用例收集到测试用例库中,经过合理的分类,供测试人员选择使用,将极大地提高软件问题的发现率。

## 3 软件测试中的测试用例复用

### 3.1 测试用例的可复用性

对于基于组件的软件工程,一般来说,除了针对某些特殊的用户需求,复用过程中的组件在结构和功能上都不太可能发生很大的变化,因此采用这些方法生成的测试用例都是可复用的。实际上,在组件的开发过程中,低层被测对象的测试用例或其部分内容常常被复用在高层被测对象的测试中,它们包括<sup>[5]</sup>:

- (1)单元测试的功能确认类用例可以在部件集成测试阶段复用;
- (2)部件确认测试的测试用例可以在单元配置组装测试和确认测试中复用;
- (3)部件确认测试的测试用例在配置项组装测试和确认测试中的复用;
- (4)配置项确认测试的测试用例在系统综合测试和系统验收测试中的复用。

### 3.2 测试用例复用

测试用例复用就是把一个软件的测试用例在新的软件测试中使用,或在软件做出修改时在新的软件测试中使

用,也包括在相似软件间进行一定的选择后使用。其中主要包括测试用例的设计思想、具体内容、执行步骤,以及测试过程中产生的信息在新的软件测试中的使用。

软件测试经过单元测试、集成测试、确认测试、系统测试后,产生了大量的测试用例和测试方法,所有的这些测试用例和测试方法都可以重新利用。整个测试过程是可以复用的,从功能测试到性能测试,许多项目的整个测试过程流程是相近的。在面向对象系统中,可复用的测试用例通常被组织成层次结构,划分为不同的测试用例组,同时还需要有一个良好的组织结构、独立于其他的环境的运行机制。尤其对专门从事软件测试的机构来说,在某一行业的软件测试中,设计出基于复用的测试用例,采用相应管理技术提供一个复用的环境,实现测试用例的复用,将有助于提高软件测试的工作效率,降低软件测试的费用,解决测试人员的经验不足等问题。

### 3.3 测试用例复用实现的思路

对第三方测试机构来说,要实现测试用例的复用,须针对特定的项目,按照测试用例的一般描述,精心设计、选择、并收集大量的测试用例到测试用例库中,对它们实施有效管理。在这个基础上,按照一定的复用原则,并采取相应复用技术,实现测试用例的复用。

首先,基于复用的目的,对所使用的测试用例进行统一的建模组织。有效地将测试用例收集到测试用例库中,并按照行业、项目等进行多级合理的分类、组织、存储。对采用不同方式描述的测试用例,将分别实现不同程度的复用。

其次,对数据库中存有的测试用例实行有效管理。通过提供有助于复用的多种查询方式,确保测试用例的复用程度。对不同类别的测试用例进行复用频率的统计,为测试人员提供有效的复用信息,在保证软件测试质量的前提下,提高了测试的效率。

最后,对数据库中的测试用例具体实现复用。通过查询出所需的测试用例,得到测试用例信息,针对不同描述方式的测试用例采用不同的复用方式进行复用。

复用的简单流程见图1。

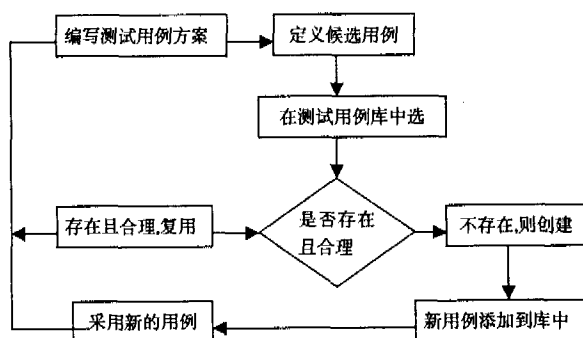


图1 复用简单流程图

### 3.4 测试用例复用的实现过程

对一个软件测试来说,测试用例复用可以划分为同一软件在不同测试阶段、不同时间测试下的复用和相似软件

之间的复用。

(1)同一软件测试。不同测试阶段的测试用例复用是指在项目开发过程中,低层测试对象的测试用例可能部分地用到高层对象的测试中,例如集成测试可能要用到单元测试的测试用例等。对于专门从事软件测试机构来说,可能要对同一个软件在不同时间下进行多次测试。按照图 1 的复用流程图,如果软件在上一次测试未通过,那么产生的大量测试用例将被存储到测试用例库中,在新的一次测试中,可以查询到相关的测试用例,直接导出运行,来对软件进行测试。测试通过后,产生的新的测试用例被有选择地存储到库中,用作以后相似软件测试的复用。

(2)相似软件的测试。首先考虑以下主要因素:软件所属的行业、软件的功能、采用的编码语言、运行的环境,以此来判断两个软件是否相似及相似的程度如何。如果可以判断被测软件和测试用例库中某一软件相似,进而在测试用例库中查询出相似软件的测试用例,对不同的测试用例进行不同程度的复用。对于采用文本方式描述的测试用例,参照设计步骤,导出后编写相应的测试用例代码,并运行;对采用编码语言编写的测试用例,可以通过测试用例库所提供的导出方法,针对被测软件,做一些类似函数名、输出形式的改动,而其核心处理过程并不发生太大的变化,最后导出并运行测试用例,测试被测软件是否通过。此时的项目间的测试用例复用主要是设计思想、测试方法、执行步骤、执行过程中产生结果信息,以及对测试用例的选择、测试用例中使用的测试数据等的复用。

### 3.5 对测试用例复用的评价

在面向对象的系统中,测试用例的设计和实现往往对应于被测对象的需求、设计和环境要求,它们缺乏统一的

结构,测试用例之间也存在着相互的联系。所以,测试用例的复用也是一定程度和一定数量上的复用,其中还与采用的复用技术、测试人员和其他客观因素相关。而对于第三方评测机构来说,主要进行验收测试,对被测软件系统进行性能、压力方面的测试,能否有效地将这些测试用例组织起来,每个测试用例能否独立地被运行,都决定着测试用例的可复用能力大小和测试效率的高低。

## 4 总 结

通过介绍软件测试过程中测试用例的设计、软件和软件测试的复用,说明了在软件测试中实现测试用例的复用的可行性、必要性,针对第三方测试机构,采用测试用例库管理的方法具体实现一定范围的测试用例复用。进而说明了测试用例的复用在保证软件质量的前提下,改进了软件测试技术,提高了软件测试的效率。

### 参考文献:

- [1] Binder R V. Testing Object - Oriented Systems[M]. Massachusetts: Addison - Wesley, 2000.
- [2] 马瑞芳. 计算机软件测试方法的研究[J]. 小型微型计算机系统, 2001, 24(12): 2211 - 2213.
- [3] 齐治昌, 谭庆平, 宁 洪. 软件工程[M]. 北京: 高等教育出版社, 1997.
- [4] 杨美清. 软件复用与相关技术[J]. 计算机科学, 1999, 26(5): 1 - 2.
- [5] 徐仁佐, 陈 斌, 陈 波, 等. 构造面向对象软件可复用测试用例的模式研究[J]. 武汉大学理工学报, 2003, 49(5): 592 - 593.
- [6] (上接第 68 页)
- [7] 的图片数明显高于文中的算法。这是由于在判断图像相似程度时加入了图片中共有对象数目和角度关系两个约束,从而使图像检索精度提高。
- [8] 4 结 论
- [9] 提出了一种表示图像中对象空间关系的方法,并将查询图像与数据库图像中共有对象数目考虑到图像的相似性度量中,使图像检索的精度得到提高。但文中的对象空间关系表示并未考虑到对象之间的相对位置关系,如何合理地将对象相对位置关系加入到文中的算法中值得进一步研究。
- [10] 参考文献:
- [11] [1] 李向阳, 庄越挺, 潘云鹤, 等. 基于内容的图像检索技术与系统[J]. 计算机研究与发展, 2001, 38(3): 344 - 354.
- [12] [2] 徐 杰, 施鹏飞. 基于内容的图像检索技术[J]. 中国图像图形学报, 2003, 8(9): 977 - 983.
- [13] [3] Chang S K. Iconic Indexing by 2D String[J]. IEEE Trans Pattern Analysis and Machine Intelligence, 1987, 9(3): 413 - 428.
- [14] [4] Chang S K. Representation and retrieval of symbolic pictures using generalized 2 - DStrings[A]. SPIE Proc Visual Communications and Image Processing [C]. Philadelphia: [s. n.], 1989. 1360 - 1372.
- [15] [5] Lee S Y. 2 - Dstring: A New Spatial Knowledge Representation for Image Database System[J]. Pattern Recognition, 1990, 23(10): 1077 - 1087.
- [16] [6] Huang P W. Using 2DC+ - String as Spatial Knowledge Representation for Image Database System[J]. Pattern Recognition, 1994, 29(9): 1249 - 1257.
- [17] [7] Nabil M. Picture Similarity Retrieval Using the 2D Projection Interval Representation[J]. IEEE Transaction on Knowledge and Data Engineering, 1996, 8(4): 533 - 539.
- [18] [8] Egenhofer M J. Point - Set Topological Spatial Relations[J]. Int' J GIS, 1991, 5(2): 161 - 174.
- [19] [9] 曹 茜. 拓扑关系渐变的定性推理[J]. 陕西师范大学学报(自然科学版), 1999, 27(3): 28 - 32.