

第 3 章

单个特性测试方法

第 2 章介绍了探索式测试思考方式和基本模型，但并没有详细讨论探索式测试的测试设计和测试执行。从第 3 章开始，本书将用 3 章的篇幅详细讨论探索式测试在功能测试上的典型技术和方法。笔者在实际的项目测试过程中，根据不同产品的特性，将产品功能测试分成三个层次，如图 3.1 所示。

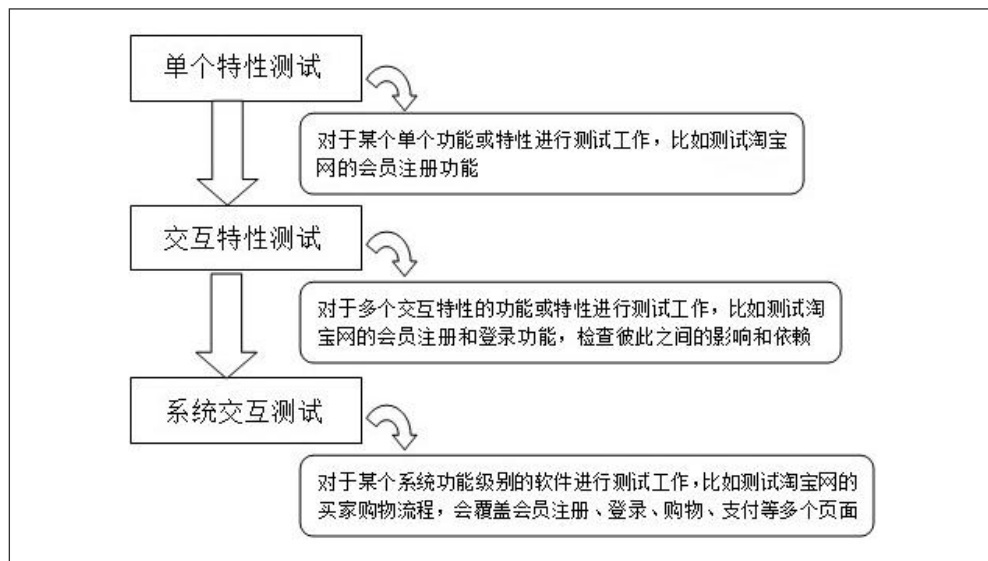


图 3.1 功能测试基本层次

本章讨论单个特性的测试方法，主要关注探索式测试在该层次上的测试设计和测试执行，并介绍能够快速对特性进行测试和分析的方法。接下来的两章将分别讨论交互特性测试和系统交互测试。

3.1 联想输入模型

软件测试的典型过程是驱动被测产品到某个特定状态，然后输入测试数据或执行特定操作，最后检查产品的输出，以侦测潜在的错误。其中，构造有效的测试输入是软件测试的关键因素之一。本节将讨论探索式测试如何构造测试输入。

Cem Kaner 对测试过程中的输入与输出类型进行了非常全面和合理的划分 [Kaner08]，请看图 3.2 中的输入参数。

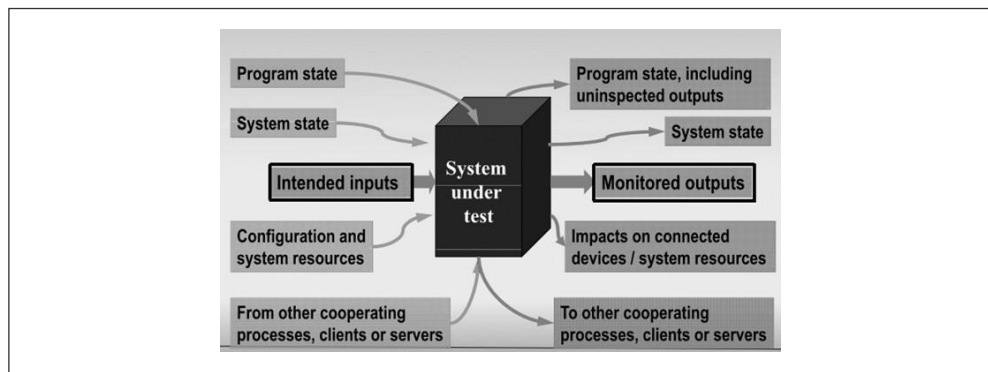


图 3.2 输入参数

从图 3.2 可以看出，测试人员可以控制的输入参数和可以观察的输出结果是众多且复杂的。一般情况下，计划输入（Intended inputs）的数据完全由测试人员控制，程序状态（Program state）、系统状态（System state）、配置与系统资源（Configuration and system resources）的取值不完全受测试人员控制，但是测试人员对它们可以施加影响并予以观察。而且，测试要想深入，必须深入到输入参数的各个部分。不过，计划输入的数据是最基础的，是测试人员需要首先掌握的，

也是本章要讨论的重点。

从测试用例的组织结构来看，一个测试用例的好与不好，在很大程度上取决于这个测试用例的输入。大家很熟悉的等价类分析和边界值分析就是选择测试输入的典型技术。探索式测试作为一种测试风格，也需要用到这些经典的测试设计方法，但是会从不一样的角度来进行组合和变化。为此，本书提出了联想输入测试模型。

在讨论具体的模型之前，本书需要介绍一下“测试模型”的概念。这是笔者特意选择的词汇，旨在明确区别于测试方法或测试技术。首先，测试模型的用意是，在一定的测试范围内建立一个模型图，以便让测试人员快速地了解 and 掌握该测试范围内的关键测试点。笔者将这些关键测试点组织起来，描述它们之间的关系，从而得到了测试模型。另外，笔者认为测试模型隶属于更高层次的测试思维，能够帮助测试人员完善功能或特性的测试设计。为了更好地理解和使用探索式测试方法，本书也将类似的测试方法的集合称为测试模型。图 3.3 详细地说明了几个概念之间的关系。

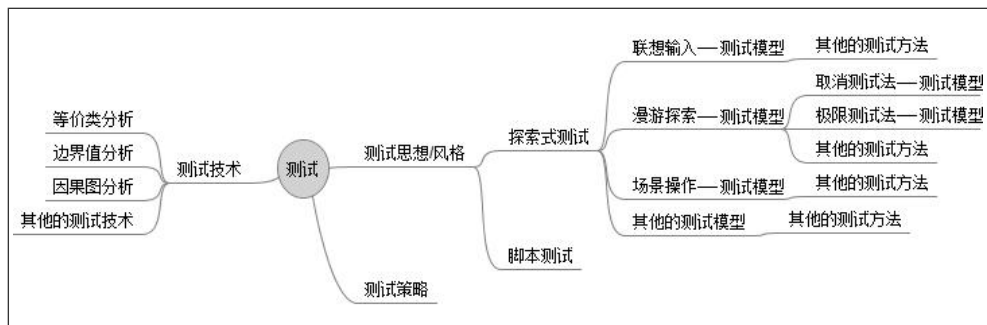


图 3.3 测试概念之间关系

3.1.1 联想输入模型介绍

联想输入模型的基本方法是，先在所有输入的全集中生成分类子集，然后根据测试对象选择一个子集进行测试，接着评估测试输入是否提供了足够的测试覆盖。如果测试覆盖有待提高，则继续选择测试自己进行测试。图 3.4 描述了子集

选择流程。

为了有效且快速地选择输入子集，笔者基于 James Whittaker 的测试理论 [Whittaker01]提出了易于理解和记忆的联想输入（Legend Input）模型。

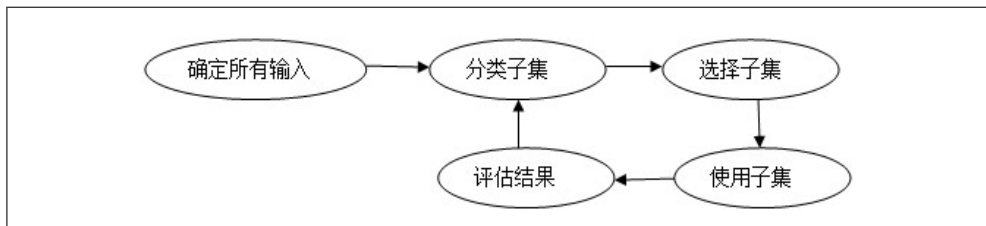


图 3.4 子集选择流程

首先看如图 3.5 所示的联想输入的基础分类，它测试输入划分为原子输入和抽象输入。其中，椭圆表示第一级的分类，矩形表示对第二级的子分类，圆角矩形表示实际测试的测试策略和注意点。

- 原子输入：对系统而言是简单到不能再简单的输入，属于单个事件，如单击按钮和输入密码 123456 等。
- 抽象输入：将有相互关联的原子输入合并在一起，则构成抽象输入，如提交用户注册表单可视为抽象输入，它包含输入用户名、密码、生日等原子输入，图 3.5 也对抽象输入的测试给出了一些建议。

图 3.6 进一步分析了原子输入的非合法输入部分，总结了软件处理非法输入的三种常见模式，并针对每种模式给出了相应的测试策略。先将原子输入进行合法和非合法输入的基础分类，然后针对输入处理代码的模式设计不同的测试策略，这其中的测试思路和策略的变化和联想是需要软件测试实践中不断总结和积累的。

- 输入筛选器：用于防止非法的输入值被传递给应用软件的功能代码，其产生的错误信息通常不返回给用户。
- 输入检查：属于功能代码的一部分，通常用 if-else 语句来实现，产生特定的错误信息。

- 异常处理代码：把整个例程（Routine）作为一个整体进行异常处理并产生通用的出错信息。

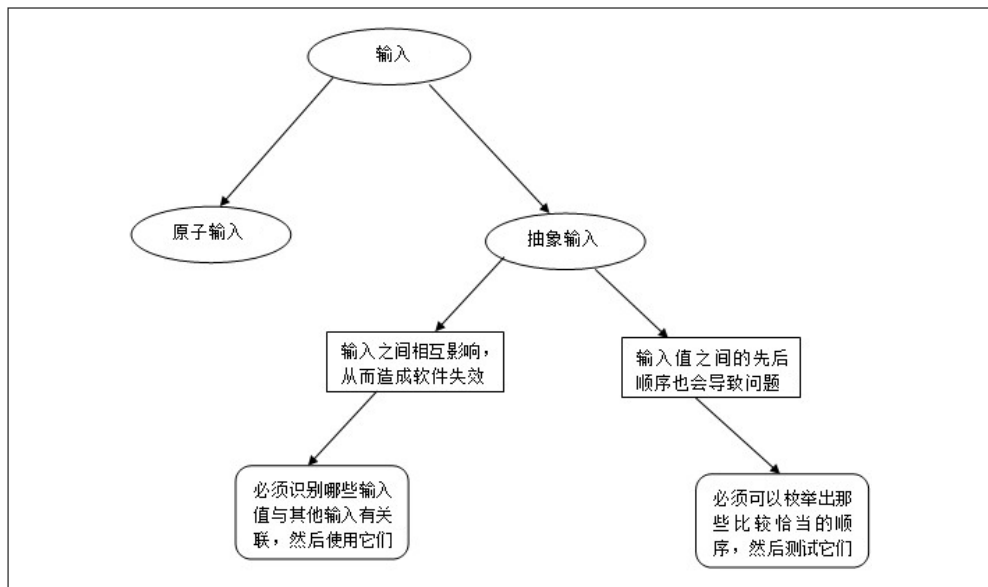


图 3.5 联想输入的基础分类

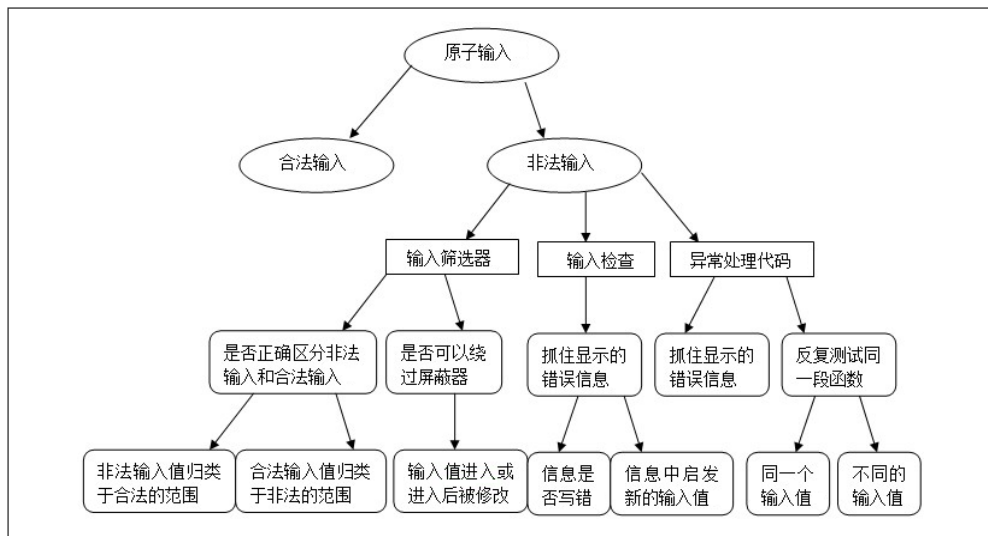


图 3.6 原子输入的非非法处理

图 3.7 分析了合法输入的情况。测试执行的时候需要考虑非常规的输入，特别是可接受输入的特殊字符。大部分特殊字符只在那些比较特殊的情况下才会出现，这些测试输入的场景往往有许多缺陷案例，测试人员需要谨慎分类。

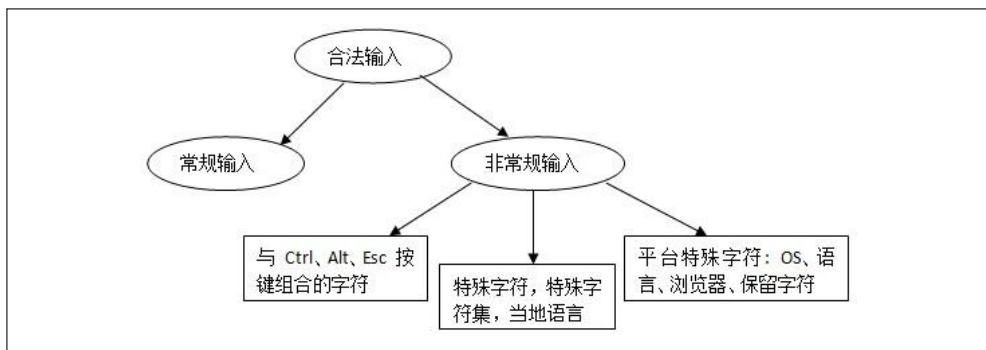


图 3.7 合法输入的分解

对于常规输入，也需要进一步细化，细化结果请参考图 3.8。其中，对于默认值的测试需要充分考虑。测试人员不但要测试输入界面的默认值，还要测试清除默认值后，输入流程能否继续。如果可以继续，空值（NULL）会不会导致软件缺陷。

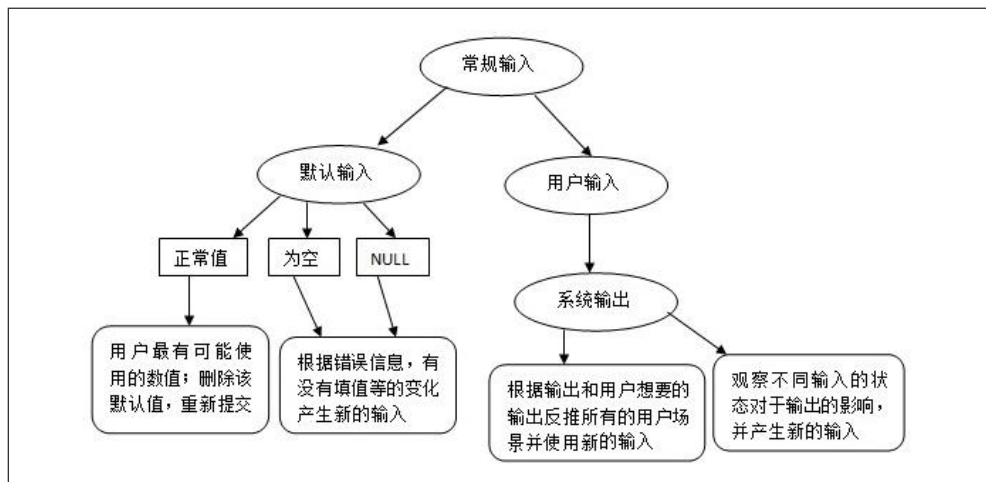


图 3.8 常规输入的分解

从图 3.8 可以看出系统输出也需要进一步细化，细化后的结果如图 3.9 所示。测试人员需要逆向思维，能够从系统输出中推算出该使用什么测试输入，从而保证系统所有的输出都是经过检查和测试的。

以上所介绍的模型图都不是一成不变的，需要各位测试同仁进行完善和补充，以使其更具有传承性和系统性。在软件测试实践过程中，如果遇到一些未能完全匹配该模型的输入或输出，需要大家在选择输入子集时，不要拘泥于模型图建议的子集分类标准。如在选择常规输入的子集的时候，可以联想到该子集与抽象输入的关系，或联想到该子集与系统输出的关系。更深刻地理解输入参数的类型及变化模型可以帮助大家更容易地联想到更好的输入数据。此外，测试人员对一个应用程序进行测试的目的不一样，使用的具体测试模型也会有所不同。

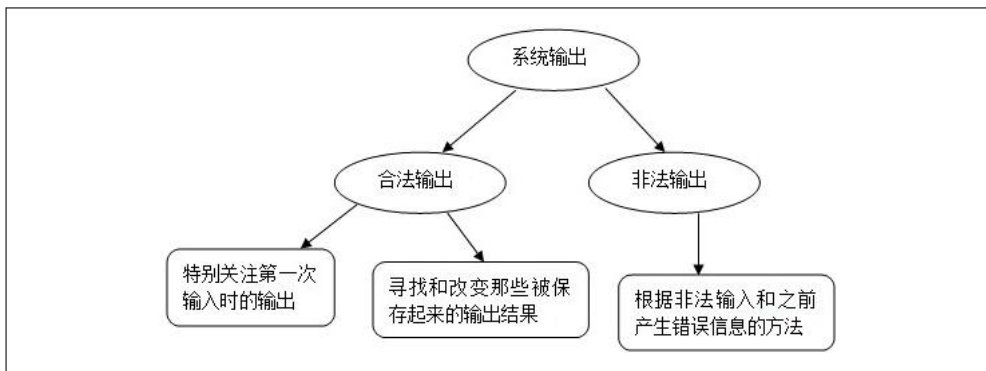


图 3.9 系统输出的分解

在功能测试中，由于测试之前无法全面掌握所有的业务功能和详细需求，所以测试人员必须站在用户的角度全神贯注地使用产品，并发现和记录所遇到的潜在问题。探索式测试在很多情况下靠使用应用程序的功能来理解隐含的业务功能和详细需求，从而完善测试前的测试设计和快速设计下一步的测试。当然，探索式测试不排除阅读文档或任何与产品相关的文献，积极地阅读需求文档也是提高探索式测试覆盖率的一个重要途径。

3.1.2 联想输入模型的实践案例

为了更好地应用前面讲述的联想输入模型，也更好地体会探索式测试的基本过程，我们来进入实战环节吧！测试对象是如图 3.10 所示的三角形分类程序。在用户输入三角形的三条边长后，该程序判断三角形的类型，可能的输出结果是：非三角形、一般三角形、等腰三角形和等边三角形。由于测试对象的功能简单，可作为单个特性来测试。

该程序由 James Bach 开发，常用于他组织的培训课程，以帮助测试人员分析在使用某个输入后，该如何选择新的测试用例来验证新的测试点。本书将其作为一个案例来初步分析如何去应用联想输入模型。测试时如果发现产品行为和预期的不一致，应用联想输入模型的测试人员应该主动地变更测试设计去深入研究产品行为（这个也是探索式测试的一个主要的特征）。

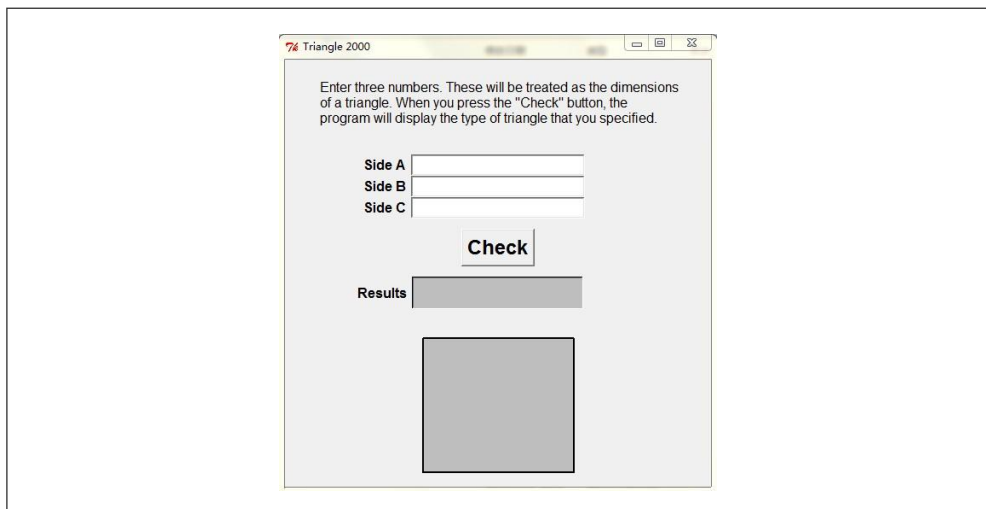


图 3.10 三角形应用程序

下面介绍如何应用联想输入模型对单个特性进行探索式测试。

(1) 要好好理解应用程序中的使用说明（Enter three numbers...），测试人员

进行探索式测试时需要站在用户的角度去使用该应用程序（确定我们了解的需求）。

(2) 分析出原子输入：Side A、Side B、Side C 和 Check Button。

(3) 选择一个原子输入项 Side A，猜测其合法输入（如果无法得到输入规则）是什么，暂时假设为大于 0 的数字。

(4) 在这些合法输入里面找出哪些是常规输入，哪些是非常规输入。对于 Side A 来说，常规输入是大于 0 的整数或其默认值（为空），非常规输入需要考虑浮点数，16 进制数字，+、-、*、/等特殊字符。

(5) 在考虑非法字符输入的时候，通过单击 Check 来运行程序，从而确定该应用程序对于非法字符是怎么处理的。这个步骤就是在探索应用程序是如何处理非法输入的。

(6) 参照模型将所有原子输入项分解完后，就需要分析出抽象输入。在本例中，就是用 Side A、Side B 和 Side C 之间的关系来判断它们是否构成三角形，如果是三角形，又是什么类型的三角形。首先要找到这些关联关系，另外还要确定每个关联关系对应的正确输出是什么。

(7) 识别出所有关联关系后，就使用抽象输入来验证这些关联后的结果。这里还可以多观察一下在选择某个原子输入的时候，其输出是什么。例如，测试人员发现 Side A 输入为 f33 或 33f 时，输出结果是一样的。于是，他在 Side A、Side B 和 Side C 都输入 f33，发现程序输出“等边三角形”。他再次尝试在 Side A、Side B 和 Side C 都输入 33，发现程序还是输出“等边三角形”。他再次变化测试输入，在 Side A 输入 33f，在 Side B 和 Side C 输入 33，发现程序还是输出“等边三角形”。这是，他可以得出一个假设：程序在处理原子输入时，会忽略数字前后的字母。然后，他可以设计更多的测试来验证或推翻该假设，从而逐步建立产品的功能模型。

(8) 再通过不同的顺序来进行 Side A、Side B 和 Side C 的原子输入，判断它们之间是否存在先后顺序关系。

(9) 找到应用程序对应的所有输出，包括非法输出。本例中，有两个输出结果：输出框 Results 和正方形的显示图。这两个结果之间的关系也要考虑，即是否有必然的一致性。列出 Results 对应的所有结果：非三角形、一般三角形、等腰三角形、等边三角形。选中某一个输出，反推出应采用的输入，如选择等边三角形，可以采用输入 1,1,1 或 222,222,222 或 1234567890,1234567890,1234567890 等。此外，还要考虑输出的持续性和间断性，一个是多次输入都产生同一个输出，另外一个就是多次输入都产生不同的输出（这种场景主要检验程序处理是否保留中间结果）。

(10) 考虑抽象输入的时候，需要不断地变化原子输入的值，这里强调多考虑最大值的表现形式。对于本例来说，可以手动输入很多数字或字符（如超过 1000 个字符），以探索原子输入 Side A 可接受的最大值。测试人员还可以利用程序生成超大的数值或超长的字符串（例如超过 10 万个字符），对程序实施极限测试。

本节粗略地介绍了如何应用联想输入模型来进行探索式测试。遗憾的是有限的文字叙述不能演示如何操作该应用程序，无法完整地应用模型测试整个应用程序，也不能很好地显示如何基于程序输出快速设计出下一个测试思路，但本案例还是给出了一些想法。希望大家能很好地理解该模型，并真正应用到测试实践中。这个模型只适用于单个较简单的功能或特性，对于有交互性的特性较多的应用程序，还需要更多的模型去支撑（第 4 章和第 5 章将详细说明）。

3.2 互联网测试模型

上一节我们描述了如何使用联想输入模型来进行单个特性的测试设计和测试执行，但存在一些问题或业务需求是不能使用该模型去解决的。

图 3.11 是淘宝网会员登录的二次验证身份的对话框，测试人员对于输入验证码就可以设计很多测试用例。但是如果只使用联想输入模型来设计，就会遗漏一个用例场景：连续输入 5 次错误的验证码后，该验证码失效，淘宝网会给出新的登录提示信息。这个遗漏场景其实是需求的一部分，测试人员要想把该需求的测试场景全部设计出来，就需要用到新的方法或模型。



图 3.11 淘宝网登录二次验证

图 3.12 展示了黑盒视角下的被测产品以及相关因素。

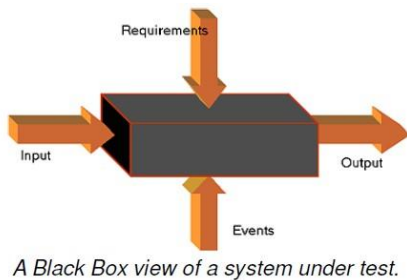


图 3.12 黑盒角度看测试

- 输入 (Input)：对于用户来说，指所有可以输入的值，如在图 3.11 的验证框中输入验证码 555555。

- **需求 (Requirement)**：输入之间的关系组成的用户使用需求，如连续输入 5 次错误的验证码后，该验证码失效。
- **事件 (Event)**：某个可以触发相关功能或特性的按钮或链接，如单击“发送验证码”按钮。
- **输出 (Output)**：产品接受输入和事件，根据需求产生输出。如输入错误的校验码，淘宝网给出提示信息“校验码错误”。

由图 3.12 可知，测试人员还需要从需求和事件的角度去设计测试用例。为此，本章将介绍互联网测试模型和漫游测试模型。这些模型可以让测试人员在测试设计和执行的时候有一些好的思路和注意点。一旦我们了解到某些特性容易存在一些类似的问题，就可以更快速地发现产品缺陷。本节所介绍的互联网测试模型是笔者从自身经验出发，取之于互联网应用，并应用于互联网应用的测试方法集，同样也“因地制宜”地做出了定制化，这种策略调整恰恰符合语境驱动测试的精神。

3.2.1 互联网测试模型来源

在目前的互联网行业，有很多通用的功能或特性，而在这些通用功能的测试过程中，会出现一些很类似的问题和风险。淘宝网作为国内知名的互联网企业，在主站功能的质量把控上都做了很多保护性的措施，也利用测试人员自己的能力为主站功能的质量保驾护航。笔者在淘宝网技术质量部工作的过程中，了解和发现了很多高级、资深测试工程师的测试经验抽象出了一些可以给测试同仁借鉴的测试模型和测试手段。

本节回顾测试模型的来历。淘宝网技术质量部在 2010 年初开启了 GBA (Golden Bug Award) 活动，专门来评选每个季度的最有含金量的缺陷，奖品为价值 1000 元人民币左右的纯金奖牌，同时也会有 8~10 个入围奖。下面简单介绍淘宝网技术质量部的 GBA 活动的主要流程。

(1) 由各个产品线的团队领导 (Team Lead) 进行组织, 将测试人员发现的好的缺陷进行标准文档化的提名, 并对提名缺陷进行自评打分, 后面将详细说明缺陷模板和打分规则。

(2) 由淘宝网测试技术委员会的成员分别负责一到两个产品线的提名缺陷的分组初评, 该委员需要了解和沟通缺陷的发现过程、业务背景和采用到的技术手段, 并给出晋级到 GBA 复评的名单, 一般来说评选分数高于 20 分的提名缺陷晋级下一轮。

(3) 活动负责人组织测试技术委员会的所有成员对于所有晋级的缺陷进行复评, 由缺陷提名者现场描述该缺陷的发现场景, 委员也可以提出自己关心的问题, 然后给每个缺陷打出相应的分数, 直到完成所有晋级到复评的缺陷的现场评审。

(4) 活动负责人根据每个成员的评分, 计算出排名前十的缺陷提名者, 并且对于排名前三的缺陷组织 GBA 终评, 确定最后入围 GBA 终评的人选。

(5) 活动负责人组织测试技术委员会的所有成员进行 GBA 终评, 同样需要入围者的参与, 通过投票和评分分析等多种手段来确定最后获得大奖的人选。

(6) 活动负责人组织每个季度的 GBA 颁奖大会, 推广和分享金奖缺陷的发现和测试手段, 也同样推广所有入围前十名的提名者, 并通过邮件或其他方式分享所有获奖缺陷的发现手段和方法。

以下是 GBA 活动的缺陷模板。

缺陷描述

[缺陷场景]:

[执行动作]:

1.

2.

[预期结果]:

[实际结果]:

缺陷原因

.....

缺陷影响

.....缺陷发现过程

.....

缺陷解决办法

.....

测试技术或方法传承

.....

表 3.1 列出了 GBA 活动的评分细则。

表 3.1 具体的标准细则

技术维度	得分	评分标准（1~5分）
缺陷发现阶段		发布阶段（1分）、执行测试阶段、编码阶段、设计阶段、需求阶段（5分）
缺陷根因分析		没有分析（1分）、简单分析并提供了线索、找到了根因、提供方案被开发采纳、解决了一类问题（5分）
缺陷发现难度		显而易见（1分）、容易发现、一般、难发现、很难发现（5分）
测试技术/方法的创新性		淘汰的、流行的、有最佳实践（应用创新）、技术创新、重大创新
测试技术/方法的复用性		不能复用（1分）、难复用、容易复用、指导小范围使用、推广到了大部分相关人员（5分）
业务维度		
用户影响（基于产品线的用户）		10%以下（1分）、30%以下、50%以下、70%以下、70%以上（5分）

续表

技术维度	得分	评分标准（1~5分）
用户体验		不美观（1分）、使用不方便、难用、不能使用、对用户造成损失或危害（如金钱、安全）
总得分		

本节花了不少文字描述了淘宝网技术质量部的 GBA 活动，笔者认为分析具体的缺陷，抽象出共性，总结出测试手段，是测试发展的重要途径，也是团队建设、个人成长的好思路。这本身就符合探索式测试的测试哲学：从工作中学习，不断地提高测试人员的水平。本节提供的流程、模板和评分规则仅供其他公司进行类似评比活动参考。GBA 评比的过程和奖品可以有不同的形式，也应该扩大获奖范围，让更多的人参与到总结、分享和学习的过程中来¹。

3.2.2 经典互联网测试模型

笔者分析了 2010 年度 GBA 的一百个提名缺陷，从而抽象出一些通用的测试模型。由于部分模型紧密依赖于淘宝技术质量部的专有技术，本书将不予讨论。本节将介绍一些相对通用的互联网测试模型，它们总结了许多很好的测试思路。在测试模型抽象过程中，笔者遵循两条基本原则。

- 原则一：测试模型不能太抽象，否则非互联网行业的测试工程师很难理解，模型应用起来也不容易。
- 原则二：测试模型不能太具体，不能和业务功能联系太紧密，否则其他领域的测试工程师受限于测试的业务场景，无法应用该模型对其他业务进行测试设计。

该互联网测试模型是基于缺陷总结出来的测试手段，也就是利用已经发现的

¹ 史亮的博客文章“测试杂感：不二过”（<http://www.cnblogs.com/liangshi/archive/2010/09/23/1833374.html>）描述了另一种形式的缺陷学习活动。

错误去检测可能再次出现的错误。由于笔者所分析的错误是有限的，该测试模型在一定程度上是不完整的，是需要所有测试同仁一起来完善的。

笔者认为优秀的测试人员能发现一些隐藏比较深的缺陷，主要有以下几个关键因素：

- 对于基本的测试设计技术的掌握达到炉火纯青的地步。
- 对于错误猜测测试方法有一定的理解和应用。
- 拥有开放思维。
- 了解用户习惯，知晓不常见的用户操作和使用情景。

其中，错误猜测是探索式测试中常见的测试手段[Erik11]。也许很多人会认为该方法在很大程度上依赖于个人的测试经验，这个说法是有一定道理的，但不意味着新加入的测试人员不能很好地使用该方法进行探索式测试。利用测试模型，新加入的测试人员能够快速开始有成效的测试，并用新获取的经验丰富测试模型。

笔者将业界非常好的测试经验总结成测试模型，并将测试模型分成三个层次。第一层是功能测试模型，即一些常用功能的测试思路的大集合，详情请参考图 3.13。

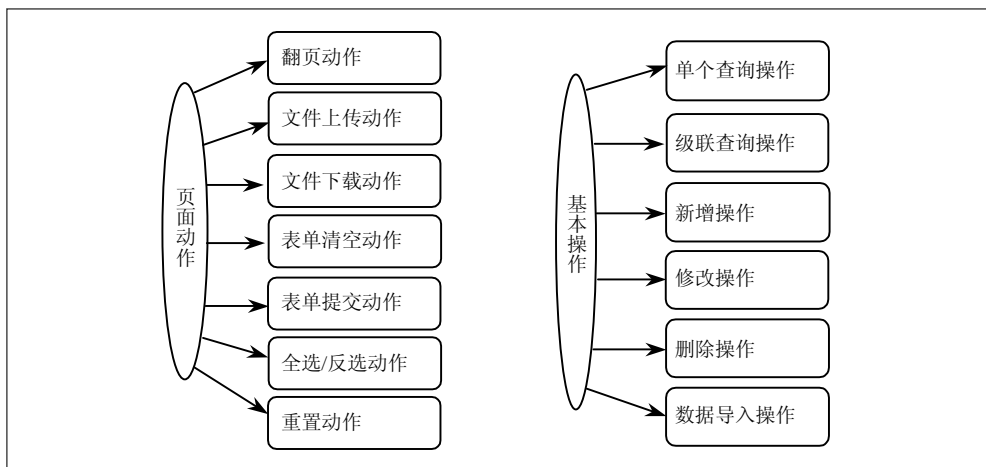


图 3.13 功能测试模型

此基础功能测试模型是初级测试工程师需要掌握的，也就是说刚进入互联网测试的新人学习基本的测试设计技术后，需要学习互联网功能测试模型，以丰富自己的领域知识和测试思路。

第二层就是线下缺陷模型，对应于本书的互联网测试模型，需要中级测试工程师甚至高级测试工程师体会和理解，同时也欢迎资深工程师完善这些测试模型和手段。

第三层就是线上故障模型，基于线上事故产生的原因进行抽象所得的模型，此模型可以进一步完善我们的测试设计思路且容易开发出新的测试方法。

下面就描述一些比较经典的互联网测试模型。

模型 1：多线程并发模型

1. 模型图（图 3.14）

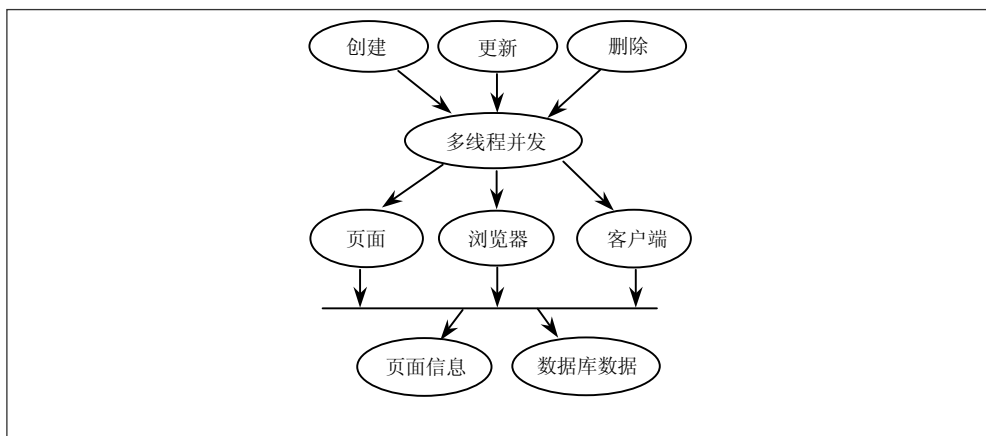


图 3.14 多线程并发模型

2. 模型解释

(1) 多线程创建、更新、删除某类数据，以多线程方式（同时打开多个页面或浏览器，或使用工具模拟）来操作数据并检查数据的完整性。

(2) 同时使用多个浏览器或一个浏览器的多个标签页 (Tab) 进行测试, 考虑 Cookie 值的变化是否影响后续的操作, 或使用场景探索模型 (下一章描述) 多角度地检查页面信息和数据库数据的正确性。

3. 应用场景

(1) 在涉及管理操作的一些基础功能的时候考虑此测试模型, 如新增一个会员、编辑一个会员信息、删除一个会员信息等。

(2) 适用于多种形式的管理功能, 测试对象不仅仅包括浏览器或客户端, 还包括其他的数据访问形式。

4. 缺陷案例²

(1) **缺陷 1:** 淘宝网会员手机注册页面, 同时注册两个相同的手机号码并提交, 后提交的注册页面出现错误提示信息。

① 发现过程:

a) 分别打开 IE 7.0 和 Firefox 3.0 两个浏览器, 每个浏览器打开一个会员注册页面, 且输入相同的手机号码, 再正确输入其他的信息。

b) 在 IE 中提交注册信息, 提交成功。

c) 在 Firefox 提交注册信息。

② **缺陷暴露:** Firefox 中的手机注册页面出现多个错误的提示信息, 其中包含样式错误。

2 本书所提到的缺陷案例已经被全部修复, 淘宝网并不存在这些问题。此外, 出于篇幅和保密的考虑, 本书未列出所有获奖缺陷。

③ **缺陷分析**：由于使用了多个浏览器同时提交，导致在 Web 页面上出现错误提示信息。

(2) **缺陷 2**：淘宝助理客户端不能正确支持多个客户端登录不同账号。

① **发现过程**：

a) 启动一个淘宝助理客户端，使用账号 A 登录，登录成功。

b) 再次启动一个客户端，使用账号 B 登录，登录后进行宝贝管理相关操作。

② **缺陷暴露**：当 A 用户未退出时，数据库连接未关闭，账号 B 的操作信息未写入本地 Firebird 数据库。

③ **缺陷分析**：Firebird 数据库不支持多进程同时访问，代码未做相应处理，淘宝助理将本地客户端所有用户登录信息存放在同一个数据库的数据表中。

模型 2：时间边界模型

1. 模型图（图 3.15）

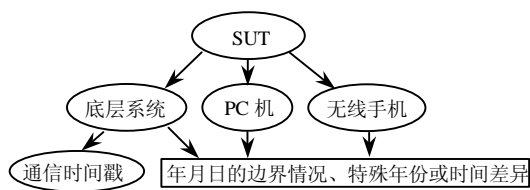


图 3.15 时间边界模型

2. 模型解释

(1) 在测试底层系统时，需要考虑时间对于产品的影响，特别是通信时间差，另外还要考虑在业务数据计算过程中年月日的边界情况、特殊年份等。

(2) 通过修改客户端或服务器时间来校验带有时间和日期限制的功能，或考虑手机和后台服务器的时间差异。

3. 应用场景

(1) 在涉及时长或时间差配置信息的功能时，先分析该时长或时间差的区间范围，然后设计边界情况和特殊值。

(2) 在涉及独立时间存储或计算的功能时，考虑这些功能交互时它们的时间是否会不一致，如果不一致发生，会不会导致严重的错误。

4. 缺陷案例

(1) **缺陷 1:** 淘宝网后台的消息中心存在跨年问题，导致用户得不到需要推送的内容。

① 发现过程:

a) 在连续的 X 天内对某一个用户一条通道推送 Y 次内容。

b) 在跨年时段（如 2010/12/30~2011/01/02）持续推送消息。

② **缺陷暴露:** 在跨年时段，产品向该用户推送的消息数量减少，这样用户数据丢失。

③ **缺陷分析:** 产品代码使用 `Calendar.DAY_OF_YEAR` 获取当前天数，在通常情况下代码可以正确工作，但遇到跨年的场景时，代码返回错误的天数，导致推送消息出错。缺陷的详细描述请参考 <http://qa.taobao.com/?p=12432>。

(2) **缺陷 2:** 进入手机淘宝网界面，无法成功打开随身购的 OperaMini 页面。

① 发现过程:

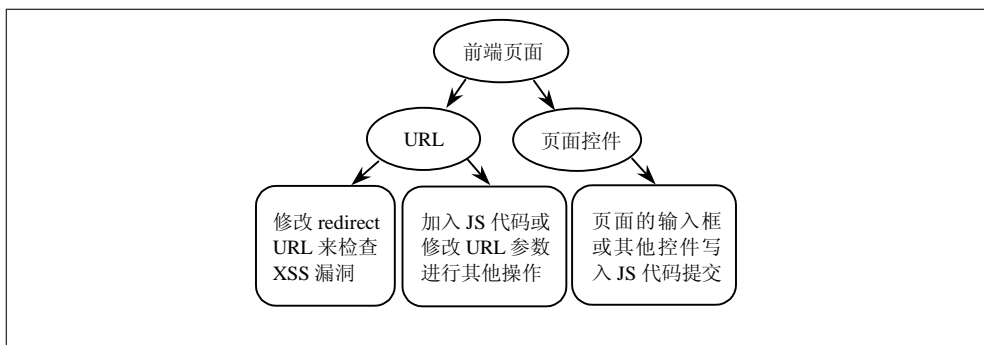
- a) 修改手机的系统时间，使其滞后于淘宝网后台服务器的系统时间。
- b) 用手机登录淘宝主页，进入“随身购”界面。
- c) 选中“推荐”选项，单击“OK”按钮。

② **缺陷暴露**: 浏览器提示“页面加载中”，当进度条加载到 80%后，再等待 7 分钟或更长时间才能打开页面。

③ **缺陷分析**: 由于客户端时间与服务端时间不匹配，功能页面加载时出现了异常。

模型 3: 页面安全测试模型

1. 模型图 (图 3.16)



2. 模型解释

- (1) 在功能测试过程中，需要考虑 URL 的安全问题和可能存在的 XSS 漏洞。
- (2) 在测试页面中输入框的校验时，考虑该页面是否存在 XSS 漏洞，可以使

用安全测试手段来做更多安全性测试。

(3) 在 URL 中加入一些 JS 代码从而实现页面 URL 跳转错误。考虑 URL 里面的参数名称和数据库中相应字段的对应关系，或修改 URL 参数以尝试访问某些本应不可操作的功能。

3. 应用场景

(1) 在测试页面链接或特性之间的跳转功能时。

(2) 当页面存在一些通用的输入控件时，特别是富文本框和一些会传送数据到后台数据库的页面控件。

4. 缺陷案例

(1) **缺陷 1:** 手机淘宝页面存在 XSS 漏洞。

① 发现过程:

a) 用户先进入到手机淘宝彩票首页，然后进入登录页面。

b) 将 URL 中的 `tpl_redirect_url` 参数值改成 `www.qq.com`。

c) 输入正确的用户名和密码，登录。

② **缺陷暴露:** 页面跳转到 QQ 首页。

③ **缺陷分析:** 该页面 URL 存在 XSS 漏洞。

(2) **缺陷 2:** 手机 Wap 旺旺消息存在 XSS 漏洞。

① 发现过程:

a) Wap 旺旺的发送方在 Wap 端发送内容为 `` 的字符串给 PC 端。

b) PC 端能正常接收。

② **缺陷暴露**: Wap 端弹出 Alert 对话框, 且消息内容为空。

③ **缺陷分析**: 手机 Wap 旺旺消息存在 XSS 漏洞, 可以被恶意用户利用盗取 Cookie 或其他会员隐私信息, 也可以在有效期内完成交易等。

(3) **缺陷 3**: 在聚划算秒杀及普通商品购买中, 用户能通过 URL 拼装绕开秒杀逻辑及时间限制进行购买。

① **发现过程**:

a) 打开淘宝网, 进入聚划算页面, 商品 A 还未开始秒杀活动, 用户进入该商品的详情页面。

b) 修改商品详情页面的 URL, 令 class 参数的取值为 `bugbtn avil J_juBuyBtns`。

c) 单击“立即购买”按钮。

② **缺陷暴露**: 页面中的“立即购买”按钮从不可点击转化为可以点击。此时, 用户可以用秒杀活动的优惠价格购买商品 A。

③ **缺陷分析**: 秒杀商品未开团前在聚划算首页展示即将开始状态, “立即购买”按钮有 5 种不同状态, 分别为: “即将开始”对应参数为 `bugbtn notbegin J_juBuyBtns`; “可购买”对应参数为 `bugbtn avil J_juBuyBtns`; “已结束”对应参数为 `bugbtn timeout J_juBuyBtns`; “还有机会”对应参数为 `bugbtn chance J_juBuyBtns`; “卖光了”对应参数为 `bugbtn soldout J_juBuyBtns`。如果 class 参数值修改为 `bugbtn avil J_juBuyBtns`, 则按钮变为可购买状态, 且页面清除了与秒杀相关的属性。

本模型涉及的 Web 安全测试是一个很丰富的主题，需要测试人员做较深入的研究，才能更周密地测试。这里的介绍只是抛砖引玉，有兴趣的读者可以参考《Web 安全测试》³等相关专著。

模型 4：功能操作异常模型

1. 模型图（图 3.17）

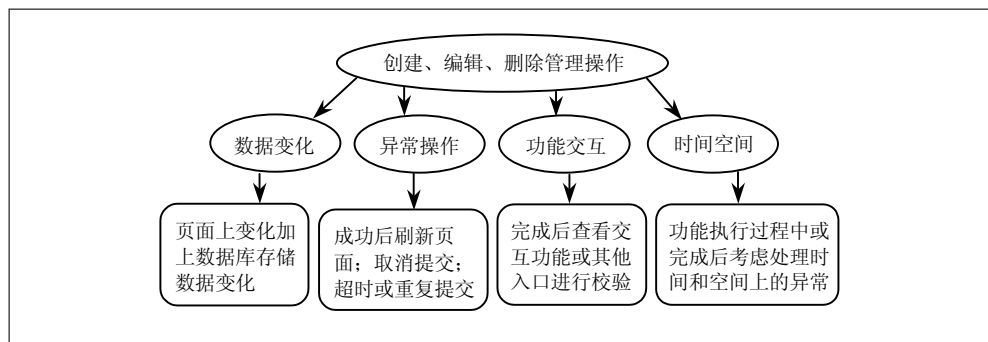


图 3.17 功能操作异常模型

2. 模型解释

(1) 编辑某类数据的时候，考虑编辑前和编辑后的数据变化，不仅检查页面上的变化，还需要查看数据库中字段值的变化。

(2) 在进行新增、更新、发布等管理操作时，操作成功后考虑刷新页面或取消提交，或考虑使 Cookie 失效，或使会话（Session）数据没有更新，或密切观察第一次提交时是否存在异常。

(3) 在修改、删除、增加数据时，考虑更新后的数据是否影响其他功能在页面上的显示。

³ <http://book.douban.com/subject/4725272/>

(4) 操作功能，检查功能执行的响应速度（或消耗的内存空间）来判断该功能是否正确执行完成，或执行过程中是否存在异常等。

3. 应用场景

(1) 在涉及创建、编辑、删除等管理操作的功能时。

(2) 当存在多个功能交互时，需要考虑在时间和存储空间上存在细微差异。

4. 缺陷案例

(1) **缺陷 1:** 淘宝商城卖家编辑图片信息，页面提示编辑成功，但实际图片存储有误。

① 发现过程:

a) 商城卖家会员成功登录淘宝网后，进入到宝贝管理页面。

b) 编辑某个宝贝信息，例如编辑非主图的图片信息，提交保存。

② **缺陷暴露:** 页面提示编辑成功，数据库中保存了新图片，但发现老的图片也仍然存在。

③ **缺陷分析:** 测试人员需要考虑编辑前和编辑后的所有数据变化，不仅仅是被修改的地方。

(2) **缺陷 2:** 淘宝卖家批量发布宝贝成功后，刷新页面导致宝贝被重新批量发布一次。

① 发现过程:

a) 卖家会员成功登录淘宝后，进入到宝贝管理页面。

b) 选择多个宝贝，进行批量发布宝贝，发布成功后，再刷新浏览器页面。

② **缺陷暴露**：宝贝被重新发布一次，发布成功后没有做页面跳转，导致再次刷新时重新提交数据，并致使宝贝重复发布。

③ **缺陷分析**：在页面提交后，没有清除表单数据，导致某些后台逻辑被重复执行。

(3) **缺陷 3**：在测试旺旺二次登录的时候，输入手机验证码后，登录时间比往常的二次登录要快（按照正常的逻辑，速度应该慢于正常的二次登录）。

① 发现过程：

a) 打开淘宝旺旺客户端，进行登录，该会员开通了二次验证保护功能。

b) 输入正确的用户名和密码进行第一次登录后，会员继续输入手机验证码进行二次登录。

② **缺陷暴露**：登录所需要的时间比往常的二次登录要快（按照正常的逻辑，由于调用更多网络，登录速度应该慢于正常的二次登录），该用户的二次验证安全保护功能失效。

③ **缺陷分析**：登录速度加快，是因为旺旺服务端调用二次验证功能失败，导致二次验证功能根本没有执行。

模型 5：搜索查询异常模型

1. 模型图（图 3.18）

2. 模型解释

(1) 进行搜索或查询功能测试时，多考虑特殊字符查询，如空格、带字母、

&、‘、“、\等特殊符号。

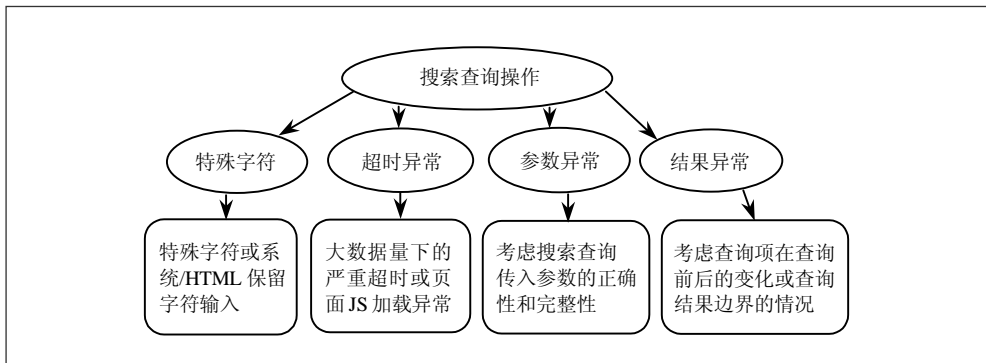


图 3.18 搜索查询异常模型

(2) 由于数据量小，在测试环境下搜索查询不会出现严重超时的问题，在预发布测试⁴的时候，注意一些搜索条件可能导致在大数据量情况下的超时问题。

(3) 对于搜索功能，查看搜索查询传入的参数的正确性和完整性，以及和搜索结果的对应关系。

(4) 在使用某些查询或搜索功能时，查询项中存在数据获取，在查询一个不存在的任何记录的情况后，再次查看该查询项的数据获取是否正确，考虑页面缓存，也同样需要考虑搜索结果中含有边界属性值的情况。

3. 应用场景

(1) 在测试搜索或查询功能时。

(2) 当存在多个查询条件和搜索条件、或这些条件存在多个交互时。

4 预发布测试是在预发布环境下进行测试，使用某一台线上服务器部署最新应用，测试数据和环境与线上产品接近或相同。

4. 缺陷案例

(1) **缺陷 1:** 淘宝网淘江湖的首页，使用公司名称搜索，返回结果不准确。

① **发现过程:**

a) 进入到淘江湖的首页。

b) 在“找到好友”中，进行公司名称的查询，且在输入内容中带空格、字母、&符号。

② **缺陷暴露:** 对于某些查询，搜索结果为空；对于某些查询，搜索结果中的公司名称排序不正确。

③ **缺陷分析:** 由于加入了一些非法字符或特殊字符，导致搜索或排序失败。

(2) **缺陷 2:** 淘宝 CRM 后台店铺留言管理，搜索店铺留言帖子时出现超时现象。

① **发现过程:**

a) 登录淘宝 CRM 后台，进入到店铺留言管理页面。

b) 搜索店铺留言帖子时，输入查询范围是过去一周，输入其他条件进行搜索。

② **缺陷暴露:** 预发布环境中店铺留言数据量较大，搜索严重超时，搜索不出结果，影响用户使用。

③ **缺陷分析:** 项目测试环境数据量较小，搜索功能工作正常，预发布环境中店铺留言数据量较大，搜索功能因为性能原因不能工作。

(3) **缺陷 3:** 手机搜索淘宝网的商品，搜索结果排序混乱。

① 发现过程:

a) 使用手机上网, 进入淘宝网, 在商城进行正常的搜索。

b) 查询项中选择按照人气逆序排序。

② **缺陷暴露:** 返回的结果集排序错乱, 没有按照人气进行排序。

③ **缺陷分析:** 通过查看查询 URL, 发现其查询字符串缺少_confp 这个人气排序的参数, 所以返回的结果集排序错乱。

(4) **缺陷 4:** 淘宝网后台用户登录仓储平台费用管理页面后, 用户仓库数据显示错误。

① 发现过程:

a) 登录淘宝 CRM 后台, 进入仓储平台费用管理页面。

b) 查询不存在的费用记录后, 单击查看页面上的用户仓库的下拉框。

② **缺陷暴露:** 下拉框显示了一个随机用户的仓库数据。

③ **缺陷分析:** 对于查询结果来说, 不仅仅关注多个搜索结果的情况, 也考虑无结果的情况。

模型 6: 数据库校验模型

1. 模型图 (图 3.19)

2. 模型解释

(1) 对于数据库中的键 (Key) 或关键字段的数据类型进行最小、最大边界的校验, 对于业务计算中的浮点数和浮点数进位进行更精确的校验。针对键字

段，多次插入或更新数据，来检验字段唯一性约束。

(2) 在数据库设计中，对于同一个字段在不同的表中的属性是否相同进行校验，重点检查“是否为空”和“限制性”这两个属性。

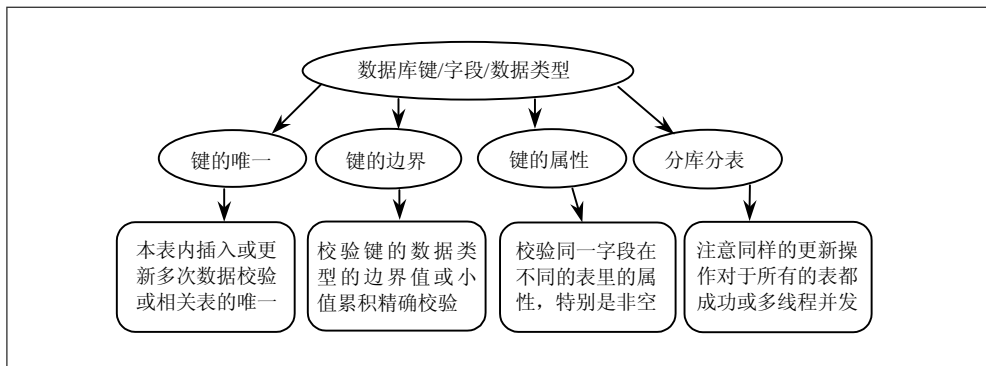


图 3.19 数据库校验模型

(3) 在数据库设计中，测试分库分表策略的操作，重点测试同样的更新操作对所有的表是否可以成功执行，还要特别留意多线程并发的情况。

3. 应用场景

(1) 在涉及数据库表设计和表的查询、更新、删除的操作时。

(2) 当存在一些对于库和表进行优化的策略的时候，考虑该策略带来的潜在的影响。

4. 缺陷案例

(1) **缺陷 1**：淘宝网购物完成后，在淘分享页面进行分享时不能成功。

① 发现过程：

a) 用户在淘宝网购买一个商品后，进入淘分享页面。

b) 将买到的商品的一些基本信息进行分享，提交。

② **缺陷暴露**：无法分享成功，主流程业务无法进行下去。

③ **缺陷分析**：当商品 ID 超过 32 位整数的最大值 (2^{32}) 时就会导致分享不能成功，原因是淘分享的相关表结构使用了 32 位整数来定义商品 ID。该缺陷表明测试人员需要检查关键字段的取值范围。

(2) **缺陷 2**：淘宝网查看商品后，收藏商品时，页面报错。

① **发现过程**：

a) 用户登录淘宝网后，选择某一个商品，并进入到该商品的详情描述页面。

b) 对于选择的商品进行收藏。

② **缺陷暴露**：偶尔会出现收藏商品不成功的现象。

③ **缺陷分析**：由于商品的 32 位 ID 废除了，导致收藏夹数据库中对应字段默认填入字符串“invalidate”，而该字段拥有唯一性约束，因此程序无法插入两条 ID 同为“invalidate”的记录。所以，当用户收藏第二个 ID 被废除的商品时，淘宝网会报错。

从缺陷中学习是测试人员的责任，学习多了，收获也就多了。希望大家一起来完善测试模型，使测试工程师能够快速地学习和理解软件测试过程中易遗漏的测试场景和异常流程。软件测试的研究方向之一是规范化描述测试模式的应用场景和使用方式，使用自动或半自动的方式匹配测试对象和测试模型。这样，测试人员就能够更快地识别风险区域，并实施有针对性的测试。

3.3 漫游测试模型

测试专家 James Whittaker 曾经在他的著作《探索式软件测试》里面提到了很

多测试方法，尤其是漫游测试方法[Whittaker09]。笔者对于这些方法做了一些分类和案例说明，特在此列出来，以便让读者掌握这些测试方法的使用场景和注意事项。

漫游测试方法总共包括二十多种测试方法，笔者将这些方法分成两种，一种是适用于单个特性的测试方法，另一种是适用于交互特性的测试方法。本节是针对单个特性的测试方法，针对些交互特性的测试方法将在下一章讨论。

3.3.1 基础测试方法

本节描述了一些基础的测试方法，新入职的测试人员可以从这些测试方法开始探索式测试的实践。

方法 1：卖点测试法（The Money Tour）

定义：测试人员找到本产品最吸引用户的功能或特性，按照产品演示步骤来测试特性。

案例：

(1) 淘宝商家的店铺发布了超级旺铺的装修模板，吸引卖家进行店铺的装修，测试人员需要找到该特性的使用步骤。

(2) 根据这些步骤进行现场的演示和使用，看是否可以得到很好的店铺装修效果，让用户真正得到实惠。

(3) 测试人员的信息来源可以是产品经理、销售人员或技术支持人员。

讨论：卖家测试法让测试人员挖掘最重要的用户情景，对于提高软件质量有如下帮助：

(1) 测试人员考察产品的“卖点”是否真正帮助用户完成任务。该“卖点”是雪中送炭的支持，还是锦上添花的修饰？尽早发现产品设计上的偏差，有助于以较低的代价开发出优质的产品。

(2) 测试人员专注于核心用户情景，将测试资源投入在用户最常用的功能和操作上，有助于提高核心功能的稳定性。

方法 2：恶邻测试法（The Bad Neighborhad Tour）

定义：测试人员找到那些缺陷数目较多的功能特性，并对邻近功能特性进行重点测试。

案例：

已有的测试表明，淘宝会员在安装支付宝数字证书后，在登录的二次验证时会遇到各种各样的问题。于是，重点测试支付宝证书和登录二次验证的邻近功能，如安装数字证书控件、安装数字证书、激活证书和首次二次验证登录等。

讨论：恶邻测试法让测试人员探索受病毒感染区域（缺陷较多的地方）的邻近功能，对于提高软件质量有如下帮助：

(1) 测试人员可以把缺陷数目同产品特性联系起来，找出缺陷较多的功能特性，可以了解产品在质量上的大致情况。

(2) 由于缺陷通常聚集出现，因此缺陷多的地方值得反复测试，另外测试人员还对邻近功能进行了重点测试，将有助于提高产品整体功能的正确性，也降低了软件质量上的风险。

方法 3：配角测试法（The Supporting Actor Tour）

定义：测试人员找到那些紧邻主要功能的特性，从而对这些特性进行额外的测试和关注。

案例：

(1) 访问淘宝网首页，登录成功后，进入我的淘宝页面，用户不仅可以看到自己曾经买过的一些商品，而且还可以看到一些辅助的链接，如查看购买此商品的会员还查看了哪些商品、列出同类商品的链接、猜你喜欢宝贝等。测试人员需要对于这些辅助的功能链接进行额外的关注和测试。

(2) 访问淘宝网首页，登录成功后，选择一个商品，单击“立即购买”按钮后，进入到确认订单页面。这时，用户可以选择送货方式。淘宝用户一般选择快递送货，测试人员可以选择其他送货方式来进行测试。

讨论：配角测试法让测试人员挖掘出紧邻主要功能的功能特性，对于提高软件质量有如下帮助：

(1) 测试人员挖掘出紧邻主要功能的功能特性，这些特性让用户能够更好地使用主要功能或给用户提供更多的选择空间。

(2) 一些有特殊需求的用户会经常使用这些非主要功能特性。测试人员重点关注这些紧邻主要功能的功能特性，将有助于提高产品的正确性和完整性。

方法 4：超模测试法（The Super Model Tour）

定义：测试人员只关心产品的界面显示，测试用户界面上的各种因素，包括用户友好性、美观性、性能等。

案例：

(1) 访问淘宝网首页，进入会员注册页面，查看页面的布局是否合理且符合用户习惯，查看按钮的大小和位置，查看各个链接是否正确，提示信息是否合理和明确，是否有错别字，对话框的内容是否正确和无歧义等。

(2) 访问淘宝网首页，登录成功后，访问淘江湖主页，查看在哪里找好友，

通过什么方式来找好友，找到好友可以进行什么类型的分享，这些页面上的显示和推荐是否符合用户习惯，刷新页面后，之前输入的数据是否丢失，是否有不安全或隐私信息显示出来，或者文字显示的颜色是否合理等。

讨论：超模测试法让测试人员更多地关注用户体验和易用性，对于提高软件质量有如下帮助：

(1) 测试人员的测试重点不是产品功能，而是用户界面。将用户界面当成超级模特，欣赏其表面是否有瑕疵，是否满足一些业界标准。

(2) 用户体验较好的产品也有利于进行探索式测试，测试人员可以更容易地访问隐秘角落里的功能特性，关注用户界面，将有助于提高产品的可操作性和易用性。

方法 5：懒汉测试法（The Couch Potato Tour）

定义：测试人员做尽量少的输入或操作流程，如接受所有默认值、保持某些字段为空、不点击相关操作按钮等，以检查程序处理默认值的能力。

案例：

(1) 访问淘宝网首页，进入会员注册页面，用户不输入任何信息，使用默认值执行提交操作。

(2) 访问淘宝网首页，登录成功后，选择一个商品，在单击“立即购买”按钮后，进入确认订单页面，不输入任何信息，进行下单操作，检查提示信息是否正确，或刷新页面后，看默认值是否被修改等。

(3) 安装淘宝的旺旺客户端，使用旺旺的添加好友功能，用默认会员名（空字符串）进行好友查找或添加；或者多次进入这个功能，看默认值是否被修改。

讨论：懒汉测试法让测试人员更多地关注产品默认值的处理，对于提高软件

质量有如下帮助：

(1) 测试人员的测试重点是系统默认值的处理能力，测试人员没做很多事情并不意味着产品也不做事情，在很多情况下，“默认的逻辑”也会执行大量的操作。

(2) 一个好的产品必须处理默认值。测试人员关注默认值处理，将有助于提高系统的健壮性和可靠性。

方法 6：反叛测试法（The Antisocial Tour）

定义：测试人员输入最不可能的数据，或已知的恶意数据，或没意义的的数据，从而检查程序的健壮性和容错性。

案例：

(1) 访问淘宝网首页，进入会员注册页面，输入最大长度的会员名或输入最大长度的密码等，提交注册信息。

(2) 访问淘宝网首页，卖家会员登录成功后，进入宝贝管理页面。发布宝贝的时候，针对宝贝的价格输入 100 万或更大的数字，或输入很小的数字，如 0.001 元、0 或负数等。在宝贝描述的输入框内输入多个含有 HTML 或 JavaScript 代码的字符串，或上传 100 张图片等。

(3) 安装淘宝的旺旺客户端，打开旺旺聊天窗口，输入超过 2KB 大小的字符串、错误的 URL、钓鱼网站网址、空格或不输入任何字符，执行发送。

讨论：反叛测试法让测试人员更多地关注产品如何处理非正常输入，对于提高软件质量有如下帮助：

测试人员的测试重点是产品处理非正常输入的能力，测试人员测试异常的输入值，关注已知的恶意输入，将有助于提高系统的健壮性和容错性。

方法 7：强迫症测试法（The Obsessive-Compulsive Tour）

定义：测试人员一遍又一遍地输入同样的数据，反复地执行同样的操作。

案例：

（1）访问淘宝网首页，进入会员登录页面，测试人员使用正确的会员名和密码进行登录操作，成功后退出，再次以同样的数据进行登录操作；或在会员注册页面，输入同样的数据进行会员注册操作，成功后再次以同样的数据进行会员注册操作。

（2）访问淘宝网首页，进入到查看一个商品的页面，将此商品加入购物车，继续选择该商品再次加入购物车；或者在淘宝网购买多个同样的商品看是否有折扣信息；或重复地对同一个商品进行下单和付款操作。

（3）访问淘宝网首页，卖家用户登录成功后，进入商品管理页面，编辑某一个商品的时候，修改商品描述信息为同一个字符，保存成功后，再次编辑同一个商品，再次修改同样的商品描述信息，重要的是使用同样的数据重复操作同样的功能。

讨论：强迫症测试法让测试人员更多地关注系统如何处理重复输入，对于提高软件质量有如下帮助：

（1）测试人员的测试重点是系统处理重复输入的能力，开发人员通常会忘记为这些场景编写错误处理程序，而这些重复输入有时候会造成重大的破坏。

（2）用户有时候会由于第一次输入错误而不得不再回头重新输入同样的内容，测试人员关注重执行的操作，将有助于提高系统的健壮性和容错性。

3.3.2 进阶测试方法

本节将描述一些进阶的测试方法，有一定经验的测试人员可以从这些测试方法中学习到更多探索式测试的实践技巧。

方法 1：极限测试法（The Intellectual Tour）

定义：测试人员对于某个特性提出很多难以回答的问题，运行程序查看结果是否正确。

案例：

（1）以淘宝的商品搜索功能为例，测试人员提出问题，如果我输入超过 256 个字符进行搜索会怎样？如果我输入不存在的商品名称进行搜索会怎样？如果我输入非法字符进行搜索会怎样？

（2）以淘宝的购物车功能为例，测试人员提出问题，如果我加入 200 个商品到购物车里面会怎样？如果我在购物车里面加入相同的商品会怎样？如果我的购物车里面所有商品价格的总和大于 100 万会怎样？

讨论：极限测试法让测试人员更多地关注系统的应变承受能力，对于提高软件质量有如下帮助：

（1）测试人员的测试重点是系统的应变承受能力，测试人员故意输入一些看似没有意义的输入，所做的一切不一定要有什么实际意义，这么做的原因只是因为系统允许这么做。

（2）这个测试法能发现各种类型的缺陷，从非常重要的缺陷到简单甚至愚蠢的缺陷。它取决于测试人员的驾驭能力，向系统提出最难的问题，就会发现系统处理困难问题的能力到底如何。

方法 2：麻烦测试法（Arrogant American Tour）

定义：测试人员对于某个特性故意设置各种障碍来看产品如何应对，思路就是采用一些非正常的使用方式。

案例：

（1）淘宝的会员注册功能要求用户在有效时间内激活账号。测试人员故意在有效时间内不激活，而在有效期过后再激活，以检查激活功能的逻辑。

（2）以淘宝的购物车功能为例，测试人员加入了多个商品到购物车里面，然后到订单确认页面，减去购物车中某个最贵的或最便宜的商品，检查订单价格的计算结果。或者新增一个最贵的或最便宜的商品到购物车中，再检查订单价格的计算结果。

讨论：麻烦测试法让测试人员更多关注产品的应变承受能力，对于提高软件质量有如下帮助：

麻烦测试法是极限测试法的变种，测试人员的测试重点也是产品的应变能力，测试人员可以故意设置各种障碍来查看系统程序是否考虑其异常性。

方法 3：通宵测试法（Clubbing Tour）

定义：测试人员连续不断地使用某个特性或将文件一直保持打开的状态，让某些特性的运行时间特别长。

案例：

（1）安装旺旺客户端，打开旺旺帮助窗口，保持该窗口一个晚上不关闭，第二天单击“提意见”标签（Tab），该标签显示 Server 500 错误页面。原因是旺旺设置了网页失效时间，只要打开应用超过了一定时间且用户未做任何操作，旺旺就会报告 Server 500 错误，且报告界面不友好。

(2) 访问淘宝网首页，卖家会员登录成功后，进入到商品管理页面。发布商品的时候，上传尺寸巨大的商品图片，检查上传操作能否顺利完成，或在编辑商品页面进行长时间的修改但不保存，在通宵操作的情况下查看页面的反应情况或 JS 的加载情况。

讨论：通宵测试法让测试人员更多地关注产品的持续运行能力，对于提高软件质量有如下帮助：

(1) 使用通宵测试法的测试人员一般会让产品一直保持运行，而不去关闭它。对于一些移动设备上的程序来说，持续运行许多天是常见的情景，此类测试就变得更为重要。

(2) 由于产品一直没有重新启动，则有些初始化数据没有被重置过，可能会发现一些潜在的问题，将有助于提高系统的可靠性和稳定性。

方法 4：测一送一测试法（The Test One Get One Free Tour）

定义：测试人员同时运行同一个应用程序的多个实例，多个用户同时使用同一个特性，类似于多线程测试模型。

案例：

(1) 同时打开两个浏览器，或某一个浏览器中的两个标签，访问淘宝网首页，进入到会员注册页面，输入同一个会员名后同时进行注册操作；或者模拟多个用户使用不同的安全登录方式同时登录淘宝。

(2) 访问淘宝网首页，卖家会员登录成功后，进入到商品管理页面，编辑商品宝贝的时候，多个用户同时编辑同一个商品，检查会产生什么样的结果。

(3) 运行淘宝助理客户端，使用账户 A 进行登录成功后，再开启一个新的客户端，使用账户 B 进行登录和并执行业务操作，查看这些操作的结果是否在数据库中记录正确。

讨论：测一送一测试法让测试人员更多地关注产品的并发处理能力，对于提高软件质量有如下帮助：

(1) 测试人员重点测试产品的并发处理能力，有助于尽早发现访问冲突、数据损坏等严重问题。

(2) 测一送一测试法来自购物时商家的“买一送一”营销策略，测试人员一旦在某个实例上发现缺陷，则在另外一个实例上也必然发现该缺陷。

方法 5：取消测试法（The Rained-Out Tour）

定义：测试人员启动某些功能操作后再停止它运行，对所有提供取消选项的功能或需要较长时间才能完成的功能执行取消操作，检查程序的自我清除能力。

案例：

(1) 访问淘宝网首页，登录成功后，进入我的淘宝页面，再进入密码修改页面，输入旧的密码和新密码及确认密码，单击提交后，有个对话框提示是否确定需要修改，并提供取消功能。测试人员需要测试该取消功能，并尝试按 Esc 键或回退按钮来取消修改。

(2) 访问淘宝网首页，卖家会员登录成功后，进入商品管理页面，发布商品的时候，上传较大图片，上传的过程中，页面显示“取消”按钮，测试人员需要测试单击该按钮后，程序是否出现异常情况，并返回到之前的页面四处地检查一下。

(3) 运行淘宝的旺旺客户端，使用账号 A 和另一个旺旺账号 B 进行聊天，在聊天窗口进行文件传送。选择传送一个非常大的文件或文件夹，在传送过程中进行取消操作，查看内存和相关特性是否存在不稳定的变化。重复多次文件传送和取消操作，检查内存和相关特性是否正常。

方法 6: 破坏测试法 (The Saboteur Tour)

定义: 测试人员试图利用每个可能的机会暗中破坏产品, 人为地创建恶劣的运行环境 (内存少、无权限、断网、故障注入等)。

案例:

(1) 使用测试工具来限制测试计算机的网速, 然后访问淘宝网首页, 会员登录后进入我的收藏夹页面; 单击“收藏我的店铺”按钮; 单击“新宝贝”链接。查看收藏的店铺里面是否正确显示该店铺上新的宝贝, 因为 JavaScript 加载速度可能引起该店铺重复显示新的宝贝。

(2) 访问淘宝网首页, 卖家会员登录成功后, 进入到商品管理页面; 单击“发布商品”按钮, 这时测试人员可以减少数据库连接池, 或断开数据库的连接, 或断开缓存的连接, 或修改缓存的存储达到最大值, 再发布宝贝, 查看结果是否出现异常。

(3) 测试人员在内存较小的机器上安装旺旺。安装完毕后, 在旺旺上开启一百个以上的聊天窗口, 观察聊天记录的显示是否很慢, 或旺旺客户端是否会崩溃。

上面列出的测试方法的部分案例是淘宝网测试工程师发现的经典缺陷。事实上在项目测试过程中很多缺陷是通过上述测试方法发现的, 只是测试工程师并没有意识到自己使用了某个特定的测试方法。所以我们有必要把这些好的测试思路总结并传承下去。对于这些测试方法, 我们需要了解使用该方法的场景, 并知晓一种测试方法只能解决某一类问题, 不能解决所有问题。所以, 有效的测试方法集不是一成不变的, 是需要所有测试同仁去完善、去创造的。

3.4 单个特性测试相关案例

接下来将介绍单个特性测试方法发现的缺陷案例, 主要以淘宝网和腾讯

QQ2011 为例。感谢董杰提供了使用探索式测试发现的 QQ2011 的缺陷案例。

案例 1：当搜索内容大于 50 个字时，淘宝旺旺的帮助中的搜索功能失效。

功能特性描述：在搜索框里输入相应的字符串，搜索帮助文档，如图 3.20 所示。

前提条件：使用 WireShark 工具截获淘宝旺旺客户发给服务端的包。

步骤 1：使用 WSocketExpert 工具抓获提交搜索的时候前端发送的包。



图 3.20 旺旺帮助中的搜索

步骤 2：修改包的长度和搜索的内容，将搜索内容长度修改为 51 个字。

步骤 3：使用伪造的包发送给服务端。

结果：发现服务端无返回结果。由于前端做了安全限制，每次发送的时候都会生成一个令牌（Token），服务端需要验证令牌后才返回结果，否则丢弃该包。

讨论：本案例可以使用“联想输入模型”来发现该缺陷。

案例 2：在旺旺帮助前台输入繁体中文字符，淘宝旺旺的后台显示出现编码问题。

步骤 1：将输入法设置为全角状态，在旺旺帮助前台输入框中输入简体中文和标点符号，后台显示正常，如图 3.21 所示。

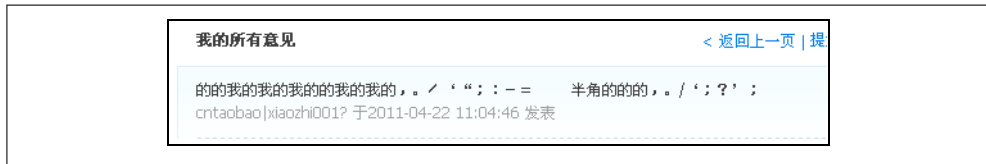


图 3.21 案例 2_步骤 1

步骤 2：将输入法设置为半角状态，在旺旺帮助前台输入框中输入简体中文和标点符号，后台显示正常，如图 3.22 所示。

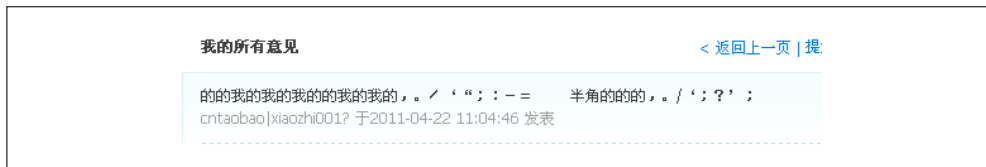


图 3.22 案例 2_步骤 2

步骤 3：将输入法设置为全角状态，在旺旺帮助前台输入框中输入繁体中文字符，如图 3.23 所示。

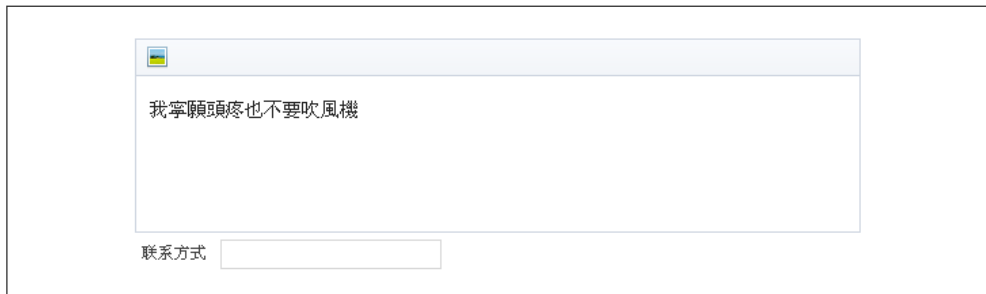


图 3.23 案例 2_步骤 3

结果：后台的结果显示有错误，繁体字符显示为问号，如图 3.24 所示。



图 3.24 案例 2_结果

步骤 4: 将输入法设置为半角状态, 在旺旺帮助前台输入框中输入繁体中文字符, 后台的结果是繁体字符显示为问号。

结果: 当在旺旺帮助前台输入繁体中文字符时, 淘宝旺旺的后台显示出现编码问题。

讨论: 本案例可以使用“联想输入模型”来发现该缺陷。

案例 3: 淘宝网会员注册时, 同步创建支付宝账号失败。

步骤 1: 访问淘宝网首页, 进入会员注册页面, 输入基本信息包括手机号码, 不勾选“同步创建支付宝”复选框, 提交后, 页面弹出提示框提示获取手机验证码, 如图 3.25 所示。

步骤 2: 发送验证码, 会员账号注册成功, 未同步创建支付宝账号。



图 3.25 案例 3_步骤 1

步骤 3: 进入会员注册页面, 输入基本信息, 包括手机号码, 勾选“同步创建支付宝”复选框, 如图 3.26 所示, 提交后, 页面弹出提示框提示获取手机验证码。

步骤 4: 发送手机验证码, 会员账号注册成功, 且同步创建支付宝账号。

步骤 5: 进入会员注册页面, 输入基本信息, 包括手机号码, 不勾选“同步创建支付宝”复选框, 提交后, 页面弹出提示框提示获取手机验证码。



您的电子邮箱： 

同意《支付宝协议》，并同步创建支付宝账户

图 3.26 案例 3_步骤 3

步骤 6：关掉弹出提示框，回到会员注册页面，勾选“同步创建支付宝”复选框，提交后，页面弹出提示框提示获取手机校验码。

步骤 7：发送校验码，再次提交。

结果：会员账号注册成功，但没有同步创建支付宝账号。

讨论：本案例可以使用“漫游测试模型”的“取消测试法”来发现该缺陷。

案例 4：QQ2011 客户端单击“重新登录”按钮后无法踢掉 Web QQ，出现同时在线的状态。

- 步骤 1：开启 QQ 客户端，QQ 用户 A 登录成功。
- 步骤 2：开启 Web QQ，QQ 用户 A 登录成功，将客户端用户 A 踢下线。
- 步骤 3：在 QQ 客户端中，马上单击弹出窗口自动登录。
- 步骤 4：这时，出现用户 A 在 QQ 客户端和 Web QQ 同时在线的情况。
- 步骤 5：在 QQ 客户端和 Web QQ 上分别选择一个好友发送聊天信息。
- 步骤 6：QQ 客户端发送成功，Web QQ 端提示网络故障失败。
- 步骤 7：过几分钟后，QQ 客户端会被自动踢掉。具体如图 3.27 所示。

分析：此处有两个矛盾。首先，根据 QQ 同一用户账号重复登录踢掉先前登录用户的规则，应该是客户端用户登录后将 Web QQ 上用户踢掉为下线状态，但是却并没有遵从以前的规则（客户端直接设置为“我在线上”时则可以踢掉 Web QQ 上的用户），最终是客户端没有把 Web QQ 上的用户踢掉为下线状态；其次，QQ

客户端和 Web QQ 上不应该存在用户同时在线状态。

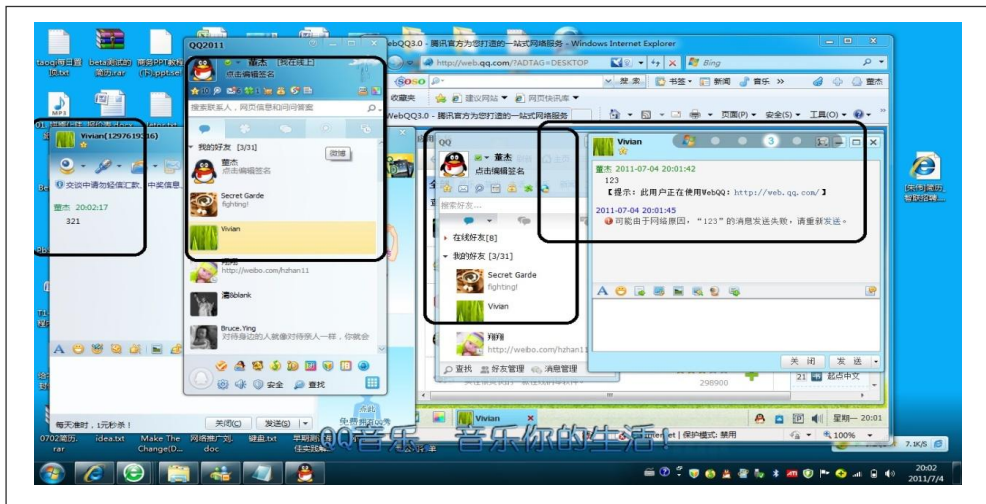


图 3.27 案例 4_步骤 7

讨论: 本案例可以使用“互联网测试模型”的“多线程并发模型”方法或“漫游测试模型”的“测一送一测试法”来发现该缺陷。

案例 5: QQ2011 主界面“个性签名”粘贴有标题文本时, 存在数据丢失。

步骤 1: 开启 QQ 客户端, QQ 用户 A 登录成功。

步骤 2: 打开 QQ 客户端主界面, 查看“个性签名”。

步骤 3: 测试人员找到存在标题的大量文本内容, 单击客户端主界面上的“编辑个性签名”。

步骤 4: 复制粘贴带有标题的大量文本内容到“个性签名”输入框。具体如图 3.28 所示。

结果: 标题粘贴成功, 其他文字没有被写入, 并且没有提供友好的异常处理信息。具体如图 3.29 所示。

霜波说测试——用户体验测试概述

用户体验测试是验证产品是否能友好和方便的满足目标用户需求的一种测试，我们设计一款产品最终的目的是希望用户能够喜欢它，使用它，最好迷恋它离不开它。用户体验测试通常集中在人机交互的问题上，比如界面是否美观，控件的使用和图标是否合理，操作是否便利，对话是否友好，语言是否本地化，安装是否方便，用户能否用最小的代价得到他想要的东西。

用户体验测试和功能测试对测试人员来说最大的不同是它是没有需求文档的，也许有的是一份用户手册，但是绝大多数中国用户的习惯都是不看用户手册的。所以我们要以目标用户的习惯去测试这个产品的易用性，美观性，甚至有时候我们要去体验这个产品的吸引力如何？

用户体验测试的方法现在常常采用的有以下三种：

1：专家测试：通常担任这一角色的都是有经验的专家测试工程师，他无需介入之前的功能测试，性能测试的任何阶段，无需了解产品的需求，设计和开发，他需要的是了解用户，需要和用户进行深入的沟通，从而知道用户最需要的是怎么样的产品。他需要的是了解现有同类的产品，了解相关产品在这一领域的使用方法和各自的特性，以及每个产品的优点和缺点是什么，从而对自己的产品提出有效的改进意见。

2：试用者：真实的目标用户是最合适的用户，也最能反馈出用户的真实想法，但是试用者的选取范围就成为一个复杂的问题，太少的试用者不能覆盖产品的各种功能，太多的试

图 3.28 案例 5_步骤 4

分析：此处有两个用户体验提升建议。首先，当粘贴的文字超出文字数量上线时，QQ 应给出友好的提示信息，同时应该存储限制之内的文字，使数据不丢失；其次，粘贴带有标题的文本信息时，QQ 也应该正常存储，使数据不丢失。

讨论：本案例可以使用“漫游测试模型”的“极限测试法”来发现该缺陷。

下一章我们将讲述多个特性交互的功能场景是如何进行探索式测试设计的。



图 3.29 案例 5_结果