

Web 应用系统中的 MOCK 测试技术

潘诗瑶¹, 黄建明²

(1. 北京邮电大学电子工程学院电子科学与技术专业, 北京 100876; 2. 北京邮电大学, 北京 100876)

摘 要: 近年来 Web 应用的市场占有率越来越高, Web 开发技术也得到了飞速的发展。然而针对 Web 应用系统的测试却不够成熟。为了提高 Web 应用的开发效率和系统的稳定性, 研制出适用于 Web 应用的测试理论与框架日益迫切。MOCK 测试技术是通过构建模拟对象实现对单元测试中的被测模块孤立进行测试的技术。而本文针对 Web 应用系统的分布式等特性, 结合 Web 开发的实际需求, 对传统软件开发中的 MOCK 测试技术进行改进, 并且提出了 Web 应用下 MOCK 自动化测试工具的系统架构, 为 Web 应用系统的测试提供了一个高效的解决方案。

关键词: 计算机应用; Web 应用系统; 软件测试; MOCK 测试

中图分类号: TP393.06 **文献标识码:** A **DOI:** 10.3969/j.issn.1003-6970.2016.12.046

本文著录格式: 潘诗瑶, 黄建明. Web 应用系统中的 MOCK 测试技术[J]. 软件, 2016, 37 (12): 214-218

The Mock Testing Technology in the Web Application System

PAN Shi-yao¹, HUANG Jian-ming¹

(Beijing University of Posts and Telecommunications, Beijing 100876, China)

【Abstract】: In recent years, the market share of Web applications is increasing, and the Web technology is rapidly developed. However, the testing of Web application system is not mature enough. In order to improve the efficiency of Web application development and the stability of the system, it is urgent to develop a test theory and framework for Web applications. MOCK test is a technique to test the tested module isolated in the unit test by constructing the mock object. This paper in view of the characteristics of distributed Web application system, combining the actual needs of the Web application development, ameliorates the MOCK testing technology in traditional software development, and puts forward the system architecture of MOCK automated testing tools in the Web application, provides an effective solution for the testing of Web application system.

【Key words】: Computer application; Web application system; Software testing; MOCK testing technology

0 引言

随着移动互联网的普及与 Web 技术的不断发展, 以浏览器作为承载平台的 Web 应用开发与解决方案越来越多。当今 Web 应用已经不仅仅满足于静态页面的展示, 而是能够独立作为一个应用系统。随着 Web 应用系统的广泛应用, Web 应用系统的测试也越来越受到人们的重视。Web 应用系统具有异构、分布、并发和平台无关等特性, 这些特性都使得对 Web 应用系统的测试要比对传统程序的测试更加困难。由于分布式系统的功能模块经常与其他模块发生耦合, 为进行单元测试引入了极大的不确定性。在此背景下, Mock 技术于 2001 年被提出并且

逐渐成为单元测试中不可或缺的技术。Mock 是指在单元测试中, 测试系统通过构造一系列符合预定义规则的模拟对象 (Mock Object) 来与被测试单元进行交互, 从而判断被试单元在正常逻辑, 异常逻辑或压力情况下能够正常工作, 返回预期的输出结果。然而在 Web 应用系统刚刚兴起不久的情况下, 没有可靠的 Mock 测试工具可用于 Web 应用场景下的 Mock 自动化测试^[1]。

本文对比了 Web 应用系统与传统软件的不同之处, 阐述了 Web 应用系统测试所面临的挑战。通过对 Web 应用系统开发所面临问题的分析, 提出了 Web 应用系统下 Mock 测试的需求, 创新性的设计了 Web 系统下 Mock 自动化测试工具的模型与架构,

作者简介: 潘诗瑶(1992-), 女, 硕士研究生, 计算机Web应用技术。

通讯联系人: 黄建明, 副教授, 主要研究方向: 多媒体技术、卫星导航技术和智能通信等。

对指导 Web 场景下的 Mock 单元测试具有重要意义。

1 Mock 测试技术

1.1 软件测试概述

互联网时代的到来与计算机技术的发展,使软件产品极大的改变了人们的工作与生活。与此同时,软件产品的质量也成为人们关注的焦点。为了避免软件出现缺陷与漏洞,从而带来巨大损失,在软件分析、设计等各个开发阶段,应对软件进行严格的技术评审。软件在编码阶段会被引入大量错误,由于人的能力有限,通过技术审查无法发现所有的错误。如果这些软件漏洞与缺陷在软件交付与投入使用时才被暴露出来,修复缺陷的成本会大幅提高,为软件开发者和使用者造成巨大的损失。

提早对软件产品进行测试可以避免上述问题的发生。软件测试就是在软件开发的初始阶段,对软件的需求、设计及编码进行严格的审查,及时的发现软件产品的缺陷,确保产品的稳定性与可靠性。软件测试就是为了发现错误而执行程序的过程^[2]。

1.1.1 软件测试模型

在当前的软件测试领域,最常见的软件测试模型是 V 模型。V 模型是 80 年代由 Paul Rock 提出,是目前使用最广泛的一种模型。V 模型是瀑布模型的变种,它明确的表明了测试过程的单元测试、集成测试、系统测试及验收测试四个不同的阶段,并且描述了这四个阶段和开发过程的各个阶段的对应关系。

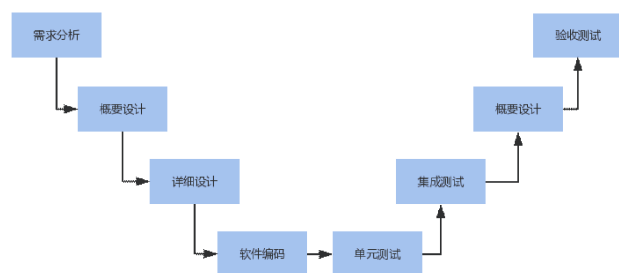


图 1 V 模型
Fig.1 the V Model

V 模型界定了测试存在于软件开发的各个阶段,并明确了软件开发各个阶段与测试各个阶段的对应关系,是软件工程中最常用的软件测试模型。

1.1.2 软件缺陷的修复成本分析

由图 1 可知,软件测试对应于系统开发的各个阶段。而随着软件开发生命周期的不断推移,修复软件漏洞的成本也不断提高。Boehm 在《软件工程

经济学》一书中曾经对软件缺陷的修复成本进行过统计与分析^[3]。在软件的各个生命周期中,引入系统漏洞及软件的修复成本如图 2 所示。

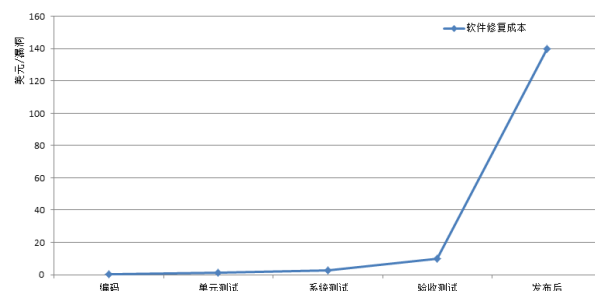


图 2 软件各生命周期的漏洞修复成本

Fig.2 The vulnerability repairing cost in the software life cycle

从图中可以看出,软件漏洞的修复成本随着软件周期的推移呈指数型增长。因此,企业和开发者应在软件开发初期,通过软件测试尽早的发现问题并对系统漏洞进行修复,避免在产品发布后造成破坏性的后果。而在软件测试中,单元测试处于软件生命的最早期,因此,对单元测试技术的研究以及单元测试工具的开发都具有重要的意义。

1.2 Web 应用场景下的软件测试

12.1 Web 系统的特性

Web 应用与传统的软件不同。Web 应用软件一般采用客户机/服务器/数据库的计算模式,在这种模式下,系统至少由三层体系结构组成:客户端(浏览器)-应用服务器-数据库服务器。可以把 Web 应用软件看成是一个使用方便、接受全局访问、具有图形化界面的大的数据库的前端。其结构示意图如图 3 所示。

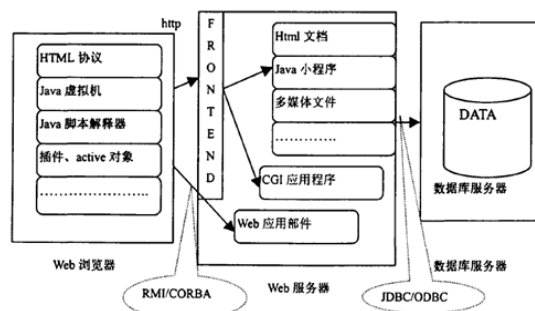


图 3 Web 应用软件结构

Fig.3 The Structure of Web Application

由图 3 可以看到,由于 Web 应用系统具有分布式的特性,使用不同编程语言、具有不同逻辑功能的单元模块之间相互依赖。这种依赖为 Web 系统的

单元测试引入了不确定性, 由于不同模块的开发进度不同, 单元测试常常无法顺利的进行。

1.2.2 Mock 测试技术

为了保证使软件最小单元之间可以无干扰的独立进行单元测试, 保证测试结果的稳定性, 就必须为软件测试提供一个细粒度的模拟对象。Tim Mackinnon, Steve Freeman 和 Philip Craig 在 XP2000 会议上的一篇文章《Endo Testing: Unit Testing with Mock Objects》中提出了 Mock objects 的概念。Mock objects (简称 Mocks) 是用来代替与代码协作对象的对象, 它为代码提供了具有特性属性和方法的对象支持。使用 Mock objects 的技术叫做 Mock 技术, 它是在单元测试基础框架基础上的扩展的代码逻辑单元测试方法。Mocks 通过隔离被测码之间、被测测试代码和测试代码之间的关联程度, 来简化测试结构, 避免被代码因测试环境出现意外, 而导致的复杂情况。它可以确保单元测试的独立性, 在一个时间段只对一个最小可测试单元进行测试, 实现软件漏洞的快速定位, 提高开发和测试效率。

2 Web 系统下的 Mock 测试工具

2.1 Mock 测试需求

由于 Mock 技术提供了与模块具有相同数据结构和功能的模拟对象, 非常适用于针对 Web 应用系统的单元测试中。使用 Mock 测试工具进行 Web 软件的单元测试, 可以非常好的解决以下场景中 Web 系统开发存在的问题。

2.1.1 客户端优先场景

在 Web 应用系统的开发过程中, 前后端分离开发已经成了主流高效的开发方式。在这种开发方式下, 服务端和客户端的开发人员可以各自独立并行进行系统逻辑、界面、接口的开发工作, 仅仅通过 HTTP 协议访问 API 进行 JSON 数据的传输来实现服务端对客户端的逻辑支持^[4]。这种模式下, 前端十分依赖来自后台的 JSON 数据, 此时若后台接口开发进度落后, 前端没有办法使用数据进行页面的填充, 会很大程度上阻碍前端页面开发的进行。

2.1.2 服务端优先场景

而当服务端接口优先完成开发时, 由于客户端开发的滞后, 无法通过构建 AJAX 对象访问服务端接口, 测试接口的准确性。若服务端接口出现漏洞, 则直到客户端 AJAX 对象开发完毕才有可能被发现, 不满足软件测试应尽早进行的原则, 同时有可能给系统带来严重的损失。

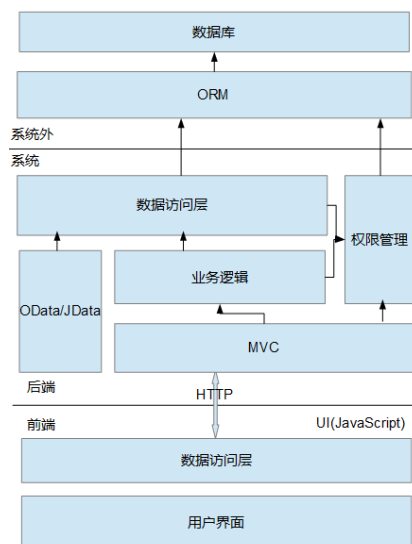


图 4 前后端分离开发模型

Fig.4 The Separation of Frontend and Backend

在这两种场景下, 引入 Mock 测试技术能够有效的解决问题。Mock 测试工具需要根据 Web 应用系统的开发需要动态构建出 Mock API, 返回与服务端具有相同数据模型的 Mock 数据, 是客户端可以脱离服务端开发进度进行 AJAX 对象的开发和测试。同时 Mock 测试工具需要模拟出客户端可能提供的请求参数, 测试服务端接口逻辑的正确性。

2.2 Mock 工具原型

在 Web 系统开发过程中, 由于服务端-客户端开发进度的不同步, 会引起相互无法为对方提供依赖对象的问题。因此, Mock 测试工具需要为客户端与服务端分别提供符合接口数据规则的模拟数据。由于 Web 系统的特殊性, 服务端与客户端通过 HTTP 访问接口进行通信, 因此, Mock 测试工具应能够通过开发者配置的项目接口文档信息, 在 Mock 测试工具所在的服务器地址动态的构建模拟接口 (Mock API), 并返回 Mock 数据。前端开发者可以访问该 Mock API 获取到与真实接口具有相同格式与含义的数据, 从而对自身的单元进行逻辑验证。经过设计的 Mock 测试工具模型如图 5 所示。

3 Mock 自动化工具的设计

3.1 业务流程

根据上一章节对 Web 系统下 Mock 测试工具的模型的研究, 可以设计出 Mock 工具的具体业务流程。开发人员应首先在 Mock 系统中录入待测项目的接口信息, 包括接口的 URI, 接口请求与请求参数的数据格式、数据类型、数据含义等参数。当接

口信息被保存时,Mock 服务器随即对该接口进行构建,即提供了运行在 Mock 服务器上的 Mock API 支持^[5]。此后,客户端人员即可访问该 Mock API 对

其最小系统进行测试,服务端人员也可通过使用 Mock 工具发送 Mock 请求,验证接口路由逻辑的正确性^[6]。完整的业务流程如图 6 所示。

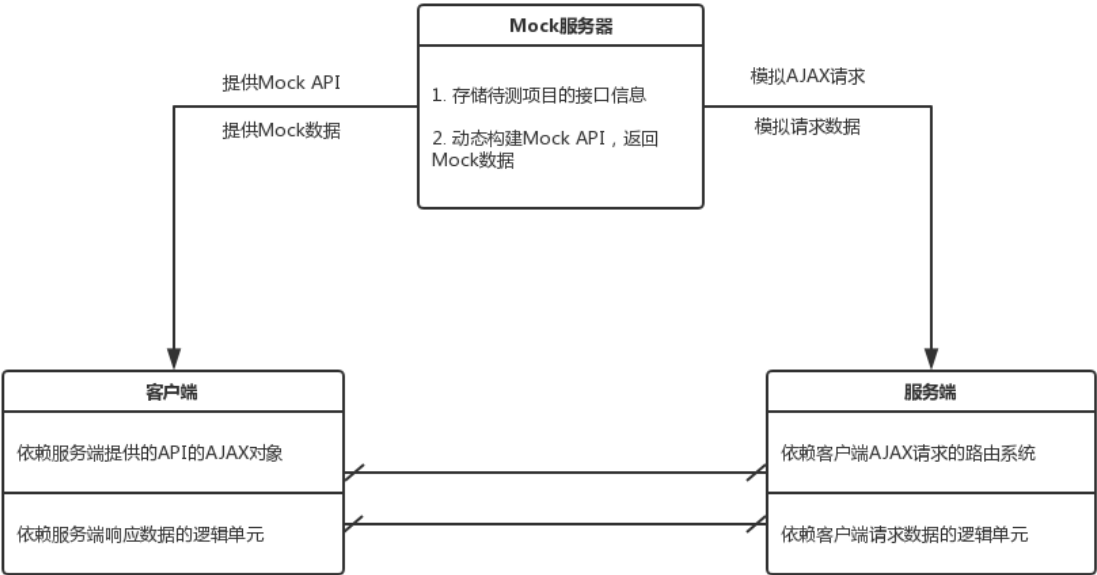


图 5 Web 系统下 Mock 测试工具模型
Fig.5 The Mock Test Model in Web System

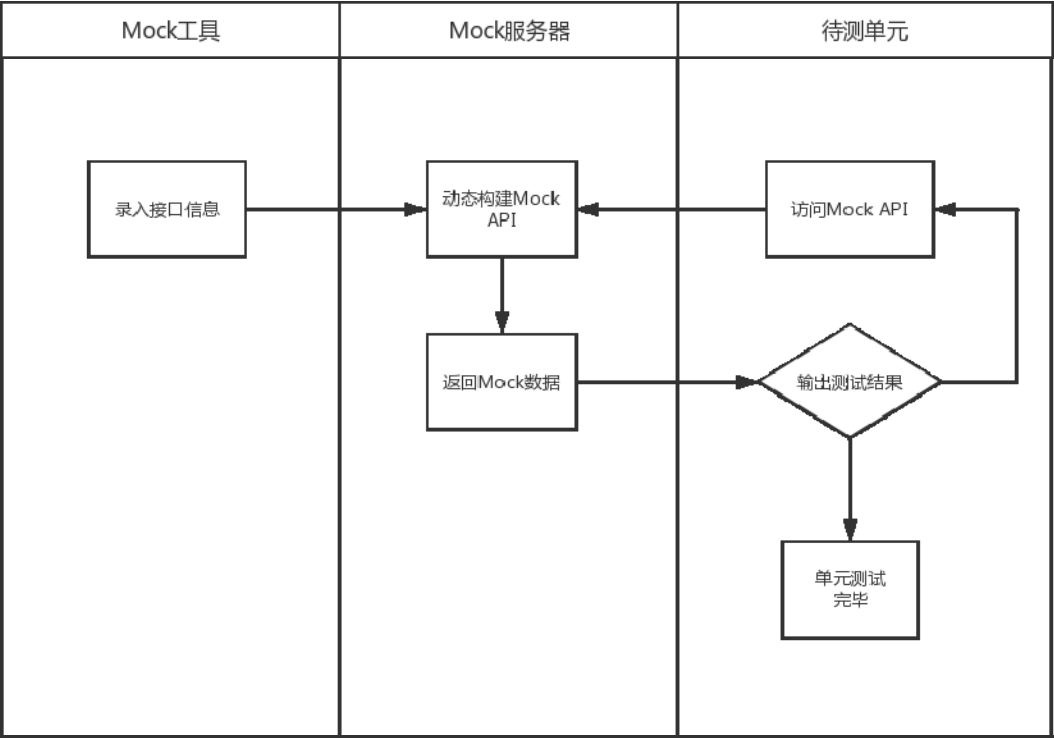


图 6 Web 系统下 Mock 测试流程
Fig.6 The Technological Process in Mock Test

3.2 系统架构

根据上一节对 Web 系统中 Mock 测试的流程设计,可以设计出 Mock 测试工具的具体系统架构。

由图 6 可以看出,Mock 工具要为开发者分别提供接口信息录入的平台界面,以及来自 Mock 服务端的 API 支持。因此,Mock 工具系统架构可以分为 Mock

平台与 Mock 服务器两部分。Mock 平台为待测试系统的接口信息录入提供入口,Mock 服务器则主要为待测试模块提供 Mock API 支持^[7]。经过设计的 Web 系统下 Mock 测试工具的最终系统架构,如图 7 所示。

开发者通过前面章节对 Web 系统下 Mock 测试

的流程与 Mock 测试工具的系统架构设计,提出了在 Web 应用系统下应用 Mock 单元测试技术,实现 Web 应用开发的前后端分离,极大的提高了开发效率和系统的稳定性,在 Web 应用开发领域具有重大的意义。

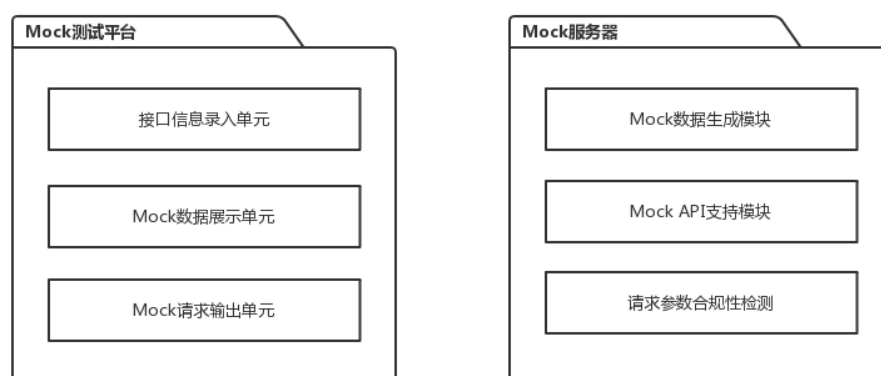


图 7 Mock 测试工具系统架构
Fig.7 The Structure of Mock Test Tool

4 结论

Mock 单元测试是针对软件测试中由于被测试单元之间存在依赖性使测试变得复杂不可预料提出的重要解决方案,在软件测试领域具有重要的地位。由于 Web 应用系统的开发和测试技术并不成熟,缺乏系统化的 Mock 测试思想及测试工具。因此,对 Web 应用场景下软件测试的研究具有重要的意义。

本文在对软件单元测试技术进行深入研究,并对 Web 应用系统测试的特殊性进行阐述之后,提出了针对 Web 应用系统的 Mock 测试思想,并根据该思想进行了 Web 应用系统具体的 Mock 自动化测试工具的流程及架构设计。该架构针对 Web 应用系统的特殊性,创新性的提出了 Mock API 与 Mock 数据的概念,能够极大的提高 Web 应用系统的开发及效率,解耦服务端-客户端间的模块依赖,实现真正的前后端分离开发。

参考文献

- [1] PATTON R. Software Testing [M]. New York: John Wiley and Sons Ltd, 2007.
- [2] 隋智泉.一种改进的单元测试JUnit框架[J]. 电脑知识与技

术, 2007(8): 479-480.

SUI Z Q. An Improved JUnit Unit Testing Framework [J]. Computer Knowledge and Technology, 2007(8): 479-480 (in Chinese).

- [3] 邓小鹏. Web应用测试技术进展[J]. 计算机研究与发展, 2007, 44(8): 74-81.

DENG X P. Web Application Testing Technology [J]. Computer Research and Development, 2007, 44(8): 74-81(in Chinese)

- [4] 张福民. 信息系统开发中的自动化测试应用研究[D]. 大连: 大连海事大学, 2006.

ZHANG F M. Automated Test Applications in the Development of Information System[D]. Dalian: Dalian Maritime University, 2006(in Chinese)

- [5] 伍向前. 单元测试中mock技术的研究和创新[D]. 大连: 大连交通大学, 2008.

Wu X Q. Mock Technology Research and Innovation in Unit Test [D]. Dalian: Dalian Railway Institute, 2008(in Chinese)

- [6] T Mackinnon, S Freeman, P Craig. Endo-Testing: Unit Testing with Mock Objects [J] Extreme Programming Examined, 2001(5): 287-301

- [7] 张敏. 面向Web应用的测试适配技术研究[D]. 山东: 山东师范大学, 2010

Zhang M. Test Adapter Technologies for Web Applications [d]. Shandong: Shandong Normal University, 2010(in Chinese)