

基于测试驱动技术的档案管理系统的开发

刘 晶

(燕山大学图书馆, 秦皇岛 066004)

摘 要:传统的系统分析设计方法难以保证档案管理系统开发的效率和质量, 将测试驱动开发的方法应用于档案管理系统开发, 可以明确档案管理系统的开发需求, 减少过度设计, 加速开发进程, 改进软件的质量。该文介绍和分析了测试驱动技术在档案管理系统开发过程中的相关应用。

关键词: 档案管理; 框架; 测试驱动开发

Development of Document Management System Based on Test-driven Development

LIU Jing

(Library of Yanshan University, Qinhuangdao 066004)

【Abstract】 The efficiency and quality of document management system development can not be guaranteed by traditional system analysis and design methods. Test-driven development methods in developing document management system can identify development requirement, avoid unnecessary design, quicken the process, and improve software quality. It introduces and analyzes the developing process application of document management system based on test-driven development.

【Key words】 document management; framework; test-driven development

1 概述

档案是一种重要的信息资源, 是人们在广泛而丰富多彩的社会实践中创造和积累起来的历史、文化资源, 档案资料因其独特的历史价值而被高度重视^[1]。档案管理既是一门专业知识, 又是一种在各类部门中有广泛应用空间的技术。但是, 由于传统的档案资料保存方式采用的是纸质、人工保存, 不仅占用了大量的库房空间, 提高管理成本, 而且档案资料在保存时极易受到损坏, 从而造成不可预测的损失。随着办公自动化的发展, 档案管理正从最初的手工档案管理发展到信息化管理, 但是目前市场上能满足档案管理的办公自动化系统少之又少。而且, 在已经使用的档案管理系统中, 大部分都是由档案室自己组织力量进行系统的开发, 都要经历需求分析、详细设计、编码、测试等多个环节, 这种开发模式周期长、耗资多、费人力, 导致档案管理系统之间缺乏协作和互通, 再加上某些系统设计需求的不明确, 影响了档案管理系统的开发质量。因此, 档案管理系统领域急需新的方法和技术来改变目前的状况。利用测试驱动开发(TDD)的先进理论和成功实践, 使用过程和系统的方法来开发高质量的档案管理系统, 有广泛的应用需求和应用前景, 具有极高的理论价值。

2 TDD 的概念和步骤

传统的系统开发中, 如果缺乏严格的过程, 在开发、发布、实施和维护系统的过程中, 可能就会碰到一些严重的问题, 其中需求的不明确可能导致很多问题。当这种情况发生时, 对系统开发的信心会动摇, 从而引起系统开发的危机, 甚至这种危机可能会比软件开发人员所面对的其他技术范围的危机更加严重、更加广泛。正是针对这一现象, TDD 方法

的应用为档案管理系统的建设展示了美好的前景。

2.1 TDD 的概念

测试驱动开发(Test-Driven Development, TDD)是敏捷开发中的一项核心实践和技术, 也是一种设计方法论。TDD的原理是在开发功能代码之前, 先编写单元测试用例代码^[2]。测试代码确定要编写产品的具体需求。TDD的基本思想是通过测试来推动整个开发的进行, 但是测试驱动开发不是单纯的测试工作, 而是把需求分析、设计、质量控制量化的过程。

2.2 TDD 的步骤

TDD 的开发步骤如图 1 所示, 分为以下 7 个步骤:

- (1)快速新增一个测试用例;
- (2)编译所有代码, 如有错误编译不通过;
- (3)做尽可能少的改动, 让编译通过;
- (4)运行所有的测试, 发现最新的测试不能编译通过;
- (5)做尽可能少的改动, 让测试通过;
- (6)运行所有的测试, 保证每个都能通过;
- (7)重构代码, 以消除重复设计。

在具体的实现上, 首先思考如何对这个功能进行测试, 快速完成针对此功能的测试用例编写, 缺少对象的测试代码会出现错误编译不通过。编译器会出现红色编译错误提示(称为红色步骤)。根据错误提示, 编写对应功能代码, 满足测试用例直到测试通过, 编译器会显示绿色编译通过的进度条(称

基金项目: 2006 年河北省档案局科研项目计划基金资助项目(2006-ky-03); 秦皇岛市 2006 年科学技术研究与发展指导计划基金资助项目(20060286)

作者简介: 刘 晶(1979 -), 女, 硕士研究生, 主研方向: 软件工程
收稿日期: 2007-04-16 **E-mail:** haner0926@163.com

为绿色步骤)。并在此过程中持续进行代码重构,使功能优化,效率提高,日益完善。然后循环进行添加功能,直到编译通过完成全部功能的开发^[3]。

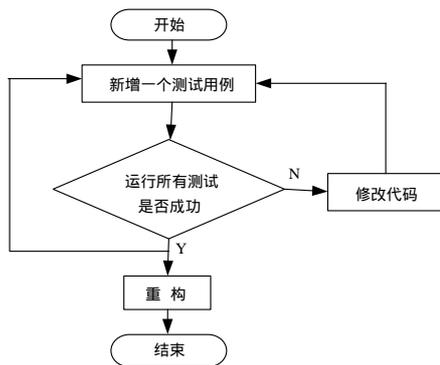


图 1 TDD 结构

3 TDD 应用于档案管理系统开发的示例及其优点

3.1 TDD 应用于档案管理系统开发的简单示例

利用TDD方法开发档案管理系统用户管理系统网上子系统一个简单的示例。系统功能为增加新用户^[4]。

(1)要测试的这个功能需求,编写这个功能的测试如下:

Public void testAddAccount(Account account);

(2)单元测试用例,增加一个新用户,编译测试代码;

(3)Account 类没有定义,编译运行测试失败(红色过程);

(4)编译器报警,缺少 Account 类定义和 AddAccount (Account account)代码定义;

(5)根据编译提示,创建 Account 类定义和 AddAccount (Account account)代码定义;

(6)重新编译,看测试用例是否运行通过;

(7)重构代码,并使得测试通过(绿色过程);

(8)循环增加其他测试用例,如测试用例 2:增加一个已经存在的用户;

(9)按照上述方法使之测试通过。测试用例 3:增加一个空用户;

(10)并按照上述方法使之测试通过。

3.2 利用 TDD 开发的优点

TDD 的重要目的不是测试软件,而是在开发过程中帮助客户和程序员去除模棱两可的需求。TDD 首先考虑使用需求(对象、功能、过程、接口等),编写测试用例框架对功能的过程和接口进行设计,而测试框架可以持续进行验证。利用 TDD 方法进行开发,程序员清楚地看到这段工作已经完成,而传统的方式很难知道何时编码工作结束。同时 TDD 可以帮助程序员全面正确地认识代码和利用代码,而传统的方式却不能达到这一目的。

事实证明,使用TDD方法开发档案管理系统,降低开发小组间交流成本,提高了相互信赖程度,避免了过渡设计。使系统可以与详尽的测试集一起发布,从而为程序的将来版本的修改和扩展提供方便^[5]。

4 结束语

测试驱动开发的基本思想是在开发功能代码之前,先编写测试代码。把这个技术的应用领域从代码编写扩展到整个开发过程。对整个开发过程的各个阶段进行测试驱动,首先思考如何对这个阶段进行测试、验证、考核,并编写相关的测试文档,然后开始下一步工作,最后再验证相关的工作。需要指出的是 TDD 的重点不是测试而是开发,其实是开发方法而非测试方法,这里驱动二字实为动词而非名词,意指:由测试驱动的、带动的开发。通过这种方法可以使档案管理系统开发需求明确,有助于加快我国档案管理系统的发展。

参考资料

- [1] 王丽华. 基于柔性制造技术的档案管理系统开发方法[J]. 现代情报, 2005, 3(3): 79-81.
- [2] Astels D. 测试驱动开发实用指南[M]. 北京: 中国电力出版社, 2004.
- [3] Beck K. Test-driven Development[M]. 北京: 中国电力出版社, 2003.
- [4] 王晓毅. TDD: 金宇塔上神像的光芒[J]. 程序员, 2006, 9(9): 80-83.
- [5] 钱静仪. 浅谈网络环境下档案管理系统资源共享[J]. 图书馆论丛, 2006, 1(1): 27-28, 60.

(上接第 278 页)

络传输、移动存储设备、打印和使用外设等方面造成的信息泄漏,提高了涉密计算机信息的保密性和可审计性。

参考文献

- [1] Gollmann D. Computer Security[M]. 北京: 人民邮电出版社, 2003.
- [2] 李培修, 敖勇, 贾永强. 内网涉密信息泄露途径及防范[J]. 计算机安全, 2005, 5(7): 75-76.
- [3] 高岩, 蒋若江. 主机防护系统中系统调用截获机制的实现[J]. 计算机工程与设计, 2003, 24(11): 76-80.
- [4] Cant C. Windows WDM 设备驱动程序开发指南[M]. 孙义, 译. 北京: 机械工业出版社, 2000.
- [5] Baker A, Lozano J. Windows 2000 设备驱动程序指南[M]. 施诺, 译. 北京: 机械工业出版社, 2001.
- [6] 朱雁辉. Windows 防火墙与网络封包机制[M]. 北京: 电子工业出版社, 2002.
- [7] Microsoft Corporation. MSDN[DB/OL]. (2003-04-07). <http://www.microsoft.com/msdn/>.
- [8] C++Builder 研究. 虚拟打印机源码 v1.0 Full Source[DB/OL]. (2006-07-13). <http://www.ccrun.com/view.asp?id=201>.
- [9] Richter J. Programming Applications for Microsoft Windows[M]. [S. l.]: Microsoft Press, 2000.
- [10] Microsoft TechNet. Regmonitro[DB/OL]. (2006-01-30). <http://search.technet.microsoft.com/search/default.aspx?siteId=1&tab=0&query=regmonitor>.