

<<基于RUP的软件测试实践>>

图书基本信息

书名：<<基于RUP的软件测试实践>>

13位ISBN编号：9787302202479

10位ISBN编号：7302202478

出版时间：2009-9

出版时间：清华大学出版社

作者：姚登峰 主编，韩玉敏 等编著

页数：396

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<基于RUP的软件测试实践>>

前言

2007年度教育部-IBM精品课程建设项目配套教材，得到了教育部和IBM中国公司的大力支持。本课程教材建设从实用角度出发，遵循ACM IEEE-CS计算机教学计划2001，注意把握学生已有的知识背景和接受能力，通过对软件企业的调研，了解到用人单位对测试人才培养的意见、需求和建议，特别是对人才知识结构、能力结构和素质等方面的要求，结合我校“双证管理型”情况（双证是指在学生毕业时既有毕业证，又有IBM课程结业证书），我们将软件测试的教学与认证考试有机地结合起来，将《国家软件评测师认证考试大纲》所要求的考试内容有针对性地纳入到教学计划中来。

合理地制定了教学大纲。

使学生完成本课程后，能熟练掌握软件测试的基本理论和基本技能，为后续课程学习和以后工作奠定坚实的基础，保证教学目标和人才培养目标的双重实现。

本书以RUP测试过程为主线，一步步引导学生思索，探究软件开发和测试实践，通过参与项目使用软件测试工具对软件进行测试，使学生掌握岗位操作技能，在实践中理解并掌握知识。

理论和实践教学在同一个项目上实现了统一，有利于学生自觉地应用理论知识解决实际问题。

突出学生在教学过程中的主体地位。

现代工业心理学研究表明，高新技术工作岗位的工作人员所需要的知识，约一半是介于经验性知识和学科理论知识之间的一种特殊的知识，即“劳动过程知识”。

软件测试在规范化的软件生产中属于软件过程工程的重要组成部分，软件测试课程构建必须使学生能在“完整工作过程”中学习。

同时注意引入在国际市场占有率高、有代表性的自动测试工具，如Mercury、IBM的有关产品，内容涵盖从安装到使用，并结合实际操作案例进行分析讲解，以此加深对概念和方法的理解，达到技术运用举一反三和知识传授与技能培养并重的目的。

按照RUP思想，RUP确定了四级测试：单元测试、集成测试、系统测试和验收测试。

这些测试级别可以是并列的，也可以是递进的，这取决于主测试计划（在项目级）和迭代测试计划（在迭代级）。

本书还是按照单元测试、集成测试、系统测试和验收测试的顺序来排列。

其中前面又加了基础部分，为学习RUP测试做准备。

后面加了实践案例，运用所学知识开展实际测试。

本书在软件测试课程内容的选取上对基本知识的建立、基本技能的培养两方面有所侧重，为学生毕业后从事软件测试职业和专业持续发展奠定基础。

总的来说，基于RUP的软件测试实践课程的教学内容分为以下几个部分：第1部分，软件测试基础：包括绪论、RUP的基础理论、RUP测试概论、手工测试与自动化测试等，即软件测试的起源和发展、测试行业的现状以及优秀测试工程师应该具备的素质、RUP的测试理论、自动化测试理论和实践、软件测试开发流程等。

第2部分，单元测试：包括测试管理、单元测试。

主要讲解测试管理的不同的阶段：组织、计划、创作、执行以及报告；测试管理所面临的挑战；IBM公司对测试管理所提的建议；实施测试管理自动化的原因；使用TestManager实现测试管理的自动化；IBM公司的单元测试工具Rational PurifyPlus。

基于RUP的软件测试实践前言第3部分，集成测试：包括组件测试和运行时分析解决方案，主要讲解组件的定义、组件测试的概念、进行组件测试的原因、组件测试方法、运行时分析原理、运行时分析分类、Test RealTime测试工具的使用。

第4部分，系统测试：包括功能测试、性能测试，主要讲解系统功能测试所面临的挑战、正则表达式、基于IBM Rational Robot的自动化功能测试框架、IBM性能测试解决方案Rational Performance Tester。

第5部分，验收测试：包括易用性和无障碍测试，主要讲解易用性和无障碍测试的基础理论、软件易用性和无障碍的一系列标准和规范、无障碍测试工具的介绍、易用性和无障碍测试的方法、无障碍测试在Web和软件方面的应用。

第6部分，案例分析：主要讲解实际案例，要求学生在梳理、总结课程体系中各项知识点和技能的基

<<基于RUP的软件测试实践>>

基础上，针对不同的开发阶段，制定相应的测试计划，设计典型测试用例，使用软件测试技术和测试工具，达到测试目标，并进行回归测试，以实现软件测试各单项专业与技能整合运用的目标。

RUP理论包含了软件开发的所有过程，因此，在学习RUP的测试技术之前，应该先对RUP的整体理论进行了解，包括RUP的特点、原则和概念等，形成对RUP的整体理论框架的基本认识，然后是对框架中整个测试体系的分析和理解。

建立框架的目的在于，学习完每个测试技术后，能够对所学到的知识有清晰的定位和明确应用的价值，并且能够在学完本教材后，对RUP的测试理论有完整的框架型理解，为继续学习RUP的其他知识打下基础，将所学到的知识填充进来，最终形成一个完整的RUP体系。

学习RUP中的几个测试基础时，可以按照工业生产用零件组装模块的惯例，由小到大地进行学习，由函数组装模块，再由模块生成程序，要学习的就是其中能保证生产正确进行的检测方法。

可以把开发过程中的函数看作是一个个小零件，由这些零件组成具有特定功能的模块，然后由模块拼接成具有完整功能的系统。

生产零件的时候，用单元测试的各种方法来对零件进行检测；把零件组装模块时，用集成测试的方法来检测模块工作是否正常；再用系统测试的方法来检测由模块组合起来的系统是否能正常工作，用性能测试等技术来检查系统性能是否达标，用无障碍测试来检验系统的可用性和关于辅助工具的易用性。

本书适合高等院校相关专业的学生及老师，也适合软件开发、软件测试人员及希望未来从事软件开发和测试的人员阅读。

本书读者应具备：具有计算机的使用经验，学过《软件工程》或者软件测试等基础知识，具有一门高级语言如C语言的基本编程基础。

为了帮助读者学习本书，作者除了提供课程网站（www.buu-testing.com）方便读者学习外，还编写了一本《基于RUP的软件测试实践题解与上机指导》，提供本书中各章习题的参考答案以及上机实习指导。

该书电子版已放在课程网站提供下载。

关于怎样学习基于RUP的软件测试，特提出以下几点看法。

1. 兴趣是最好的老师 如何看待软件测试工作，有人说软件测试很有趣，麻烦堆里有快乐，也有人说它太枯燥。

诸多说法表现一个人对这项工作的喜好。

但笔者要说的是，如果对软件测试工作没有兴趣和热情很难做到持久。

笔者最近参与了一个软件测试项目，在测试团队中，有三位是软件相关专业的本科或研究生在读学生，基础都还不错。

但是，只有一位表现最突出，因为他对软件测试抱有浓厚的兴趣，很珍惜这份社会实践的工作机会，做事认真，找出了很多高优先级的Bug。

另两位同学，在参加项目不到1个月后就以各种理由退出了。

笔者在与他们的交流中，其中一位同学说测试工作太枯燥，没有挑战性，他更希望做软件开发工作。

这位同学喜欢做挑战性的工作无可厚非，但他缺乏对软件测试技术最基本的了解，其实软件测试工作同样具有挑战性。

这位同学在7天的测试工作中，只找到了3个Bug（正常情况下，一般测试人员每天能找到5个Bug）。

因此，在绩效评比中他的成效最低。

另一位同学虽然愿意做软件测试，但是他觉得现在的黑盒测试太简单，学习不到测试技术的高级技巧，他更愿意学习白盒测试，能够自己测试软件源代码。

而现在他所做的项目没有这部分内容，所以尽管他工作成绩也不错，但是热情不大。

因此，建议同学们在学习或找工作之前，首先需要了解自己对软件测试是否有兴趣或培养起兴趣，是否热爱软件测试工作。

只有热爱和专注于某项事业时，才会做出连自己都感到吃惊的成绩来。

2. 巩固测试知识基础 练武术需要先练“蹲马步”，否则直接学习刀枪棍棒等十八般武艺，只能学到几招皮毛。

<<基于RUP的软件测试实践>>

武林高手都是基础牢固，内功深厚。

做软件测试也是这个道理。

如果认为学完本书，就可以成为测试高手，这是错误的。

本书的初衷是为初学者提供一条学习测试实践的捷径，试图将软件测试实践涉及的理论讲述清楚，降低软件测试实践的门槛，引领读者进入基于RUP的软件测试实践大门。

师傅领进门，修行在个人。

要想成为测试高手，还需要个人的努力。

个人尽可能多地参加软件测试项目，在实践中学习技能，积累经验，不断分析和总结软件开发过程中可能出错的环节。

这样，一名优秀的测试工程师就从软件测试的实践中脱颖而出了。

学习本书需要读者有良好的测试知识基础。

RUP软件测试中主要的测试自动化的质量完全依赖于测试案例和测试数据本身的质量，如果要设计应用于自动化测试的数据，了解划分等价类、确定边界值等测试基础知识是很有必要的。

常见一些初学者还没有学会测试的基本概念，就盲目地学习各种大型商业自动化测试软件，结果花了很多时间，只学会了工具的具体操作，而没有实际参与测试的能力。

到了实际测试项目中，也无法有效利用工具解决实际测试问题。

实际上，测试新手大部分应该从手工功能测试开始起步，只有成为测试高手之后才有能力使用大型自动化测试软件。

另外，测试工具的操作是很简单的技术问题，关键是如何发挥测试工具的作用，这需要了解测试的原理，并通过原理来应用测试工具。

高级的测试人员除了需要精通测试技术，还应掌握行业知识，可以提供行业软件的测试和质量保证方案。

对于初学者，要认识到经过不断努力，才可以成为测试行业专家。

千里之行，始于足下，目前最重要的是从测试入门知识开始。

所以，初学者要老老实实地学习有关测试的基础知识，学习各种测试术语、测试概念、测试分类、测试的流程、测试项目的执行过程等。

如果连这些都不懂，今后的职业发展会受到限制。

学习测试知识没有捷径，需要从一点一滴学起，日积月累，需要勤奋，需要思考，需要总结提高。

3. 为什么要学习基于RUP的软件测试给大家讲个小故事。

有个英国人学煮鸡蛋，开始，他把鸡蛋放到开水里煮时总会炸裂。

他为此想了各种方法，并找到了一个解决方案：在鸡蛋上打个孔。

但在鸡蛋上打孔带来了另一个问题：孔打小了，鸡蛋还会裂；孔打大了，蛋清会在它凝固以前流出来。

于是，这个英国人给一批鸡蛋分别打了各种不同孔径的洞，并记录下每个鸡蛋孔径的大小。

通过这一方法，他找到了一个最合适的大小——既避免了炸裂，又保证蛋清不会流出来。

这时，虽然煮鸡蛋炸裂的问题解决了，但这个英国人仍然不知道煮多长时间才能把鸡蛋煮熟。

为了解决这个问题，他又开始尝试煮不同时间的结果，并从中找出最佳时间。

最后他终于摸索到煮鸡蛋的最佳流程和方法。

这个小故事对很多中国人来说，可能只是觉得是件可笑的事例，或许认为这个英国人过于迂腐，因为聪明的中国人早就知道把鸡蛋放在冷水中与之一起加热至鸡蛋浮起来就可以了。

从煮鸡蛋这样一个小小的事例上，体现了中国人和英国人两种完全不同的思维习惯——中国人凭借自己的经验和聪明直奔结果，而英国人却把每一个过程细化后，框定为任何人任何时候都可操作的流程，然后大家都可按照这个流程执行。

现代产品的开发和生产需要制定一个标准的流程，这个流程并非凭空臆想，它如同英国人煮鸡蛋一样，是经过无数次的探究而得来的。

只有制定极为详尽的工业产品生产流程规定，每一步如何做，应该达到何种标准，才能真正地将流程应用于工业生产。

<<基于RUP的软件测试实践>>

因为参与产品生产流程的工人有成百上千，这么多工人只有遵循一套标准的流程，才能生产出许许多多相同的标准产品。

工业产品可以在出厂前设置质量检测来保证质量，但软件毕竟不同于工业产品，它只能在开发过程中监测产品质量。

在生产过程中进行质量检测比生产完毕后再进行检测的方法要先进得多，也是保证产品质量的需要，它应该对生产全过程进行追踪。

这种方法的思想正是软件过程的质量保证的精髓所在，基于RUP的软件测试就是这样的过程质量保证思想。

在欧美国家，如果去应聘软件开发者职位，招聘者就会问：“你懂RUP吗？”

如果说“不懂”，他就会让你回家去学RUP，然后再来应聘。

中国大概还没到这个程度，不过笔者相信很快中国的研发者和经理们也会看到RUP的重要性的。

4. 大学生能学好RUP理论吗国内有一种误解，认为RUP及其配套软件工具主要用于大型软件开发，学习难度大，不适合初学者。

其实RUP并不是高深莫测，只要有学习的基本条件，例如具备计算机、相关软件知识，还要有学习的耐心和毅力，是可以学好RUP的。

而且想要进入软件开发这个行业，也应该从RUP开始学起。

从RUP中可以学到管理的基本常识、分析设计、体系结构设计、测试，在.NET、J2EE、C++或其他平台上的实现，还可以学到进度管理、版本控制、分布式运算等。

因为UML是软件开发的通用语言，而RUP则蕴涵着许多软件开发者应该知道的知识。

并且RUP是一个稳固的基础，在这个基础之上，可以开发各种各样的软件系统，既可开发小型项目，也可以开发像国防系统、大型银行系统这样庞大的项目。

所以，建议有志于从事软件开发和软件测试的，都应该学习RUP。

有一个学生问笔者，国内最流行是XP (Extreme Programming, 极限编程) 方法，RUP不适合中国国情，为什么不学XP反而要学RUP呢？

其实这个问题就跟讨论C++和Java谁好的性质是一样，不能笼统地说哪个好，而且RUP不适合的，XP也未必会适合。

刻舟求剑这个成语故事大家知道，如果把RUP或XP的一整套东西生搬硬套，最后会导致项目失败。

正所谓只有最适合的方法，没有最好的方法。

笔者认为XP其实就是小型的RUP。

如果要开发一个小型项目，只有很少的团队人员，并且要在比较短的时间内完成，就可以并且应该使用XP这种轻量级的方法。

这种方法更加灵活，迭代周期更短，但这并不意味着它与RUP相对立。

实际上随着项目的增大，团队的成长，XP也可以转变成为RUP。

两者的确有差异，但这种差异也是因为不同项目的需要而造成的。

笔者认为建模是非常重要的，但XP却不这么认为。

当我们觉得应该建模的时候，XPer们就已经开始编码了。

当然，这是符合XP的要求的，的确也获得了很大的成功。

不过，这也把XP限制在小型项目的范围内。

如果在大型项目中这么做，系统很快就会陷入混乱的。

当然，这并不是XP的错，因为XP本来就只是用于小型项目的。

目前，RUP应用于小型软件企业有一定的难度，而关键在于如何根据项目需要进行裁减。

而且使用RUP需要掌握一定的技巧，正如一个高明的铁匠用铁锤可以打出一把好兵器，如果是普通人搞不好还会砸到自己的脚。

5. 加强学习行业知识 建议大家利用一切可以利用的时间学习，多阅读测试书籍，关注和游览测试网站和论坛。

要根据自己的知识基础，有选择性地阅读测试书籍。

对于初学者应先从基础知识学起。

<<基于RUP的软件测试实践>>

正式出版的书在质量和内容上都有保证，而有些测试网站和论坛的文章良莠不齐，有的只有只言片语，有的还存在一些错误。

因此，需要有一定的鉴别能力，否则会被误导，浪费时间。

对于新进入公司的员工，公司一般都要经过短暂的培训，发一些培训材料，这是今后从事工作的最好的第一手材料，针对性和实用性都很强。

它是公司测试工作经验的总结，也是今后要用到工作中的一些基本的测试知识和技术的介绍。

另外还可以借助测试项目的测试文档学习，包括测试计划、测试用例、测试缺陷数据库，可以先看看以前发现了哪些Bug，这些Bug是怎么被发现的，有什么规律和特征，学习别人怎么写测试缺陷报告。

测试人员除了学习和掌握测试技术外，还需要不断学习行业知识，这是普通测试技术人员与测试行业专家的区别之一。

学习什么行业知识呢？

根据测试的软件的应用领域决定。

例如，如果正在测试的是电信行业的应用软件，那么需要学习电信行业知识，包括术语、业务和行业技术。

怎么学习呢？

可以与客户交流，与开发人员交流，看专业书和文章。

学习行业知识是个不断进步的过程，每个行业都有很系统的知识架构。

6. 测试人员要学会思考 测试是个技术工作，要学会主动思考。

测试问题错综复杂，需要自己分析问题的性质，尝试各种解决方法，搜索网上的资料，如果实在解决不了才向测试主管求助。

测试人员如何思考？

要根据碰到的问题现象来思考，是属于测试专业知识不足引起的，还是测试用例等测试文档模糊、错误引起的；是个别现象，还是测试项目的其他内容都存在的普遍现象。

测试要从模拟用户使用的角度来看，因此要从最终用户的角度来查找问题，分析问题可能导致的严重程度。

在询问最终的解决方法前，必须根据自己的经验尝试了各种解决方法，并且尽量把发现的问题和想法告诉测试主管，证明自己已经主动思考了，只是没有找到更好的解决方法，或者不能确定自己的方法是否可行。

最后我想借用RUP理论创始人之一Ivar Jacobson大师的一句话来送给读者：“我从心底里真诚地告诉中国的开发者：尽快去学习RUP！

因为这将是大势所趋。

从RUP中，你将可以学到很多很多非常有用的知识。

在晚上、在周末、在等女朋友的时候、在喝茶的时候，总之，抽出一切时间来学习RUP。

这样的努力必将得到回报。

这就是我能给中国的朋友们最好的建议”。

在此，由衷地感谢多年来关心支持本书作者的各位朋友。

北京大学杭诚方教授、北京联合大学李启隆教授和毛世春教授等前辈给予了有力的支持和指导，并抽出宝贵的时间审阅此书。

郝增茹、崔桓祥、刘文红、蒋雪峰、张鹏、李妍、李明等老师也给予了极大的关心和支持。

感谢IBM大学合作部程郁佳、王立女士，IBM信息无障碍研究中心覃玉梅经理等多年来全力支持和帮助作者在计算机教育和计算机普及领域所从事的工作。

感谢试用本教材的一些院校，感谢他们提出了一些宝贵的意见。

本书由姚登峰主编，韩玉敏、邱云峰、黄海瑞、李娟参与了部分章节的编写，邱云峰还负责外文校译。

另外，黄海瑞和陈林波参与了程序调试工作。

由于作者水平和时间有限，难免有缺点和不足，热切期望得到专家和读者的批评指正。

希望本书能促进测试人员基础水平的提高，也为更多的人群加入测试队伍提供辅助向导。

<<基于RUP的软件测试实践>>

如果读者在学习中遇到了难以理解和解决的问题，可以访问本课程网站，也可以直接和作者进行联系。

本书属于原创，但作者的学习成长离不开网络和书籍。

在编写此书过程中，作者更是置身于现实的学术氛围，无疑要吸纳和借鉴专家和同行们先进的学术思想、方法，在此深深地感谢他们给予作者的启迪和帮助。

<<基于RUP的软件测试实践>>

内容概要

本书介绍了RUP的特点、原则和概念及RUP的四级测试（单元测试、集成测试、系统测试和验收测试）。

在内容的选取上对基本知识的建立、基本技能的培养两方面有所侧重，让学生形成对RUP的整体理论框架的基本认识，为学生毕业后从事软件测试职业和在专业上的持续发展奠定基础。

本书适合作为高等学校计算机及相关专业的本专科生教材

<<基于RUP的软件测试实践>>

书籍目录

第1部分 软件测试基础 第1章 绪论 1.1 引言 1.2 错误是不可避免的 1.3 软件测试历史 1.4 软件测试模型的演变 1.5 软件测试类型 1.6 软件测试工具的发展 1.7 当今测试行业状况 1.8 测试角色 1.9 职业规划 习题与思考 第2章 RUP基础 2.1 RUP的发展史 2.2 什么是RUP 2.3 RUP的特点 2.4 RUP软件开发生命周期 2.5 RUP过程的静态结构 2.6 RUP中的最佳软件实践 2.7 RUP中的关键原则 2.8 RUP4+1视图 2.9 RUP裁剪 2.10 实践经验 2.11 小结 习题与思考 第3章 RUP测试概论 3.1 软件测试 3.2 RUP软件测试流程 3.3 质量保证 3.4 测试团队与角色 3.5 RUP四级测试 3.6 RUP测试解决方案 3.7 RUP使用技巧 3.8 小结 习题与思考 第4章 手工测试与自动化测试 4.1 手工测试基础 4.2 自动化测试基础 4.3 测试自动化技术 4.4 测试脚本技术 4.5 自动化测试实践 4.6 自动化测试的优缺点 4.7 小结 习题与思考 第2部分 单元测试 第5章 测试管理 5.1 什么是测试管理 5.2 测试管理的内容 5.3 开展测试管理 5.4 传统测试管理的挑战 5.5 基于RUP的测试管理经验 5.6 测试管理的自动化 5.7 TM的使用 5.8 小结 习题与思考 第6章 单元测试 6.1 单元测试基础 6.2 单元测试策略 6.3 单元测试工具实践 6.4 小结 习题与思考 第3部分 集成测试 第7章 组件测试与运行时分析 第4部分 系统测试 第8章 系统功能测试 第9章 性能测试 第5部分 验收测试 第10章 易用性测试 第11章 无障碍测试 第6部分 案例分析 第12章 测试案例附录A UML基础: 统一建模语言简介附录B 测试评估摘要附录C WCAG 1.0的14条指导原则参考文献

<<基于RUP的软件测试实践>>

章节摘录

插图：好处：团队生产力，更好地结合商业需求与软件系统的开发和运作。

模式：激励团队成员做得最好。

分析师、开发人员和测试人员的跨职能合作。

管理不断演进的工件和任务，通过集成环境来加强对协作、进度和质量的了解。

确保业务、开发和运作团队作为一个集成的整体高效地工作。

反模式：培养英雄个体并为他们配备强大的工具。

软件是由才华横溢、积极上进的人员通过紧密协作创造出的。

许多复杂的系统要求一些具有不同技能的项目涉众进行协作，而大型项目通常会跨越地理和时间的界限，从而进一步增加了开发流程的复杂性。

这就是为什么人员问题和协作（一些人称之为软件开发的“软”元素）会成为灵活开发团队的主要侧重点。

遵循此原则要求回答以下问题，包括：· 如何激励人们做得最好？

· 如何在同处一地与分散的软件团队中进行协作？

· 如何在负责业务、软件开发和IT操作的团队之间进行协作？

有效协作的第一步是激励团队中的个人做得最好。

自我管理团队的概念已在项目团体中得到普及，它的基础是使团队承诺负责应交付的成果，然后由团队决定和判断影响交付成果的所有问题。

当人们意识到需要真正负责最终结果时，会更主动地去保质完成工作。

如RUP的灵活性声明所陈述的：“为受到激励的人员构建项目。

为他们创造环境，支持他们的需要并相信他们能完成工作。

”第二步是鼓励跨职能协作。

如walkerRoyce所说的“软件开发是一项团队工作。

”迭代方法更需要团队密切配合工作。

我们要拆除分析人员、开发人员和测试员之间通常存在的壁垒，图2-9拓宽这些角色的职责以确保在快速变化的环境中进行有效协作。

每个成员都需要了解任务和项目远景。

在团队发展到一定规模时，要提供有效的协作环境。

这些环境使度量值收集和状态报告变得更为方便和自动化，并使围绕着配置管理的构建管理和日志自动化。

这种效率可减少会议次数，使团队成员将更多时间用在具有更高生产率和创造性的活动上。

这些环境还应通过简化交流、使处于不同地区和时区的团队成员能够沟通，以此达到更有效的协作。

这些环境的示例包括从共享项目房间到联网或基于web的解决方案，例如集成开发环境、配置和变更管理环境。

<<基于RUP的软件测试实践>>

编辑推荐

《基于RUP的软件测试实践》：教育部IBM高校合作项目精品课程系列教材

<<基于RUP的软件测试实践>>

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>