Broadview
www.broadview.com.cn

测试实践丛书

China Software Testing Center
中国软件评测中心
作品系列

软件生命周期质量保证与测试

中国软件评测中心 组编　张向宏 主编
何伟起 等　副主编

由一线测试专家编写，供未来测试专家阅读！

電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

测试实践丛书

China Software Testing Center
中国软件评测中心
作品系列

软件生命周期质量保证与测试

中国软件评测中心 组编　张向宏 主编
何伟起 陈渌萍 周波 杨少华 王晓芹 副主编

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

随着软件规模的日益增大，软件质量问题日显突出。本书从软件生命周期的角度着重介绍软件质量、质量保证和软件测试的基本概念和技术方法。首先介绍了软件生命周期、软件质量等一般性的概念，针对软件生命周期的各个阶段的质量度量问题，提出相应的质量度量指标和度量方法，然后对软件质量保证的定义、工作过程、主要任务等进行阐述，并从软件生命周期的各个阶段分别论述了如何进行软件的质量管理，最后提出软件测试是保证软件质量的有效手段。本书除了介绍软件测试理论及当前前沿测试技术外，还对软件生命周期的各个阶段提供的测试类型进行了详细论述，并提供了大量的典型测试实例，便于帮助读者分析掌握软件质量保证的测试手段。

本书主要面向从事软件质量保证、软件开发和软件测试领域的技术人员和管理人员，以及任何对软件开发、软件质量保证和软件测试等各种实践感兴趣的人员阅读。同时，还可以作为软件测试培训课程的参考教材。

图书在版编目（CIP）数据

软件生命周期质量保证与测试 / 张向宏主编；中国软件评测中心组编. —北京：电子工业出版社，2009.5
（测试实践丛书）
ISBN 978-7-121-08561-1

I. 软… II. ①张… ②中… III. 软件开发 IV. TP311.52

中国版本图书馆 CIP 数据核字（2009）第 042369 号

责任编辑：葛娜
印　　刷：北京智力达印刷有限公司
装　　订：北京中新伟业印刷有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编 100036
开　　本：787×1092　1/16　印张：27.25　字数：593 千字
印　　次：2009 年 5 月第 1 次印刷
印　　数：4000 册　　定价：55.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：（010）88258888。

前　言

本书编写背景

软件产业作为信息产业的核心和灵魂，在促进国民经济和社会发展信息化中具有重要的地位和作用，是国家的基础性、战略性产业。随着软件规模的日益增大，软件质量问题在近年也日益突出，由于软件存在质量缺陷而引发的事故频频发生，它不仅会造成软件交付延期、开发成本递增，甚至会威胁到生命和社会安全。这再一次向我们提出了警告，必须立即重视和认真解决软件质量问题。

软件生命周期是软件的产生直到报废的生命周期，包括问题的定义及规划、需求分析、软件设计、程序编码、软件测试和运行维护 6 个阶段。事实上，软件生命周期的各个阶段都会涉及质量问题。通常来说，软件开发过程中的缺陷发现得越晚，所需付出的代价就越高。必须提前对软件生命周期的每个阶段进行质量管理，通过测试手段实现各个阶段的质量保证。

本书既有软件质量领域的理论与实践，又有软件测试领域的理论与实践，从整个软件生命周期的角度，把软件质量保证与软件测试结合起来，实现软件质量的提升。本书对于刚进入 IT 领域的软件质量保证人员和软件测试人员具有理论的指导意义和实践的借鉴意义。对于有一定工作经验的人士来说，本书知识面广，也是一本提升境界、扩展思路的宝典。

本书特色

本书逻辑层次清晰、知识面涵盖广、理论与实践结合紧密，使读者能够迅速掌握软件质量保证与测试的相关理论与实践方法。本书的特色如下。

1．强调对“过程”的软件质量保证

本书强调“软件生命周期”，分别从软件生命周期的各个阶段去分析与论述软件的质量保证与测试，纠正了国内对软件质量保证“重结果轻过程”的认识。

2．加强理论与实践的紧密结合

本书对于软件生命周期质量保证与测试的论述，不仅有深入的理论分析，还有实际项目的实践经验分享；不仅介绍了软件生命周期的质量度量、质量管理、质量保证、测试实施，还详细介绍了典型测试案例，包括功能测试、性能测试、可靠性测试等多个领域，都是一些实际的项目经验，具有很大的借鉴意义。

3．详细解析软件生命周期的测试实施

本书对于软件生命周期的测试实施进行了十分详细的论述，在软件生命周期的每个阶段，对于测试的对象、测试的策略、测试的方法及一些测试的工具都有

详细的阐述。

4．本书的思路地图

问　题	答　案
需要理解的软件质量与软件测试的基础知识有哪些	软件质量概述（第 1 章） 软件测试（第 5 章）
如何在软件生命周期各阶段进行质量度量	软件生命周期的质量度量（第 2 章）
如何在软件生命周期各阶段进行质量管理	软件生命周期的质量管理（第 3 章）
如何在软件生命周期各阶段进行质量保证	软件质量保证（第 4 章）
如何在软件生命周期各阶段进行软件测试	软件生命周期的测试实施（第 6 章）
有哪些典型的测试案例可以借鉴	典型测试实例（第 7 章）
目前软件测试技术的发展动态如何	前沿测试技术介绍（第 8 章）
实际测试实施中会用到哪些测试工具	常用测试工具介绍（第 9 章）

5．你能从本书得到什么

- 提升全局的眼界，从软件生命周期各个阶段去考虑各种问题；
- 增加知识的积累，围绕“软件质量”，丰富各个方面的知识；
- 了解软件生命周期各阶段的测试工作，加深测试的思想；

……

关于本书作者

中国软件评测中心成立于 1990 年，作为国内最早开展测试工作的国家级计算机软件产品质量检测机构，具有国家级软件质量保证专家和一支多年从事软、硬件质量保证的技术队伍，承担了多项国家“八五”、“九五”、“863”、“十五”重点攻关课题，近二十年来，一直致力于软硬件测试、电子政务评估、信息工程咨询与监理、资质认证等领域的研究与实践，在软件测试理论及实践中积累了深厚的功底和丰富的经验。

主编张向宏，现任中国软件评测中心常务副主任。在 2002 年至 2008 年期间，受国信办委托，承担中国网站调查及“振兴软件产业行动纲要”、“国家电子政务总体框架”、“关于我国电子政务建设指导意见”、“全国电子政务全过程绩效评估指标体系和方法研究”等前期研究与规划工作；负责国家质检总局政府网站、上海政府网站、青岛政府网站、武汉政府网站、常州政府网站的功能设计及内容规划等；在担任中国软件评测中心常务副主任及赛迪监理公司总经理期间大胆改革，创新机制，整合中国软件评测中心测试理论和经验，使公司经营业绩连续增长，进一步提高了公司在业界的竞争能力。

目　录

CONTENTS

软件生命周期质量保证与测试

目 录

CONTENTS

软件生命周期质量保证与测试

目 录

CONTENTS

软件生命周期质量保证与测试

目 录

CONTENTS

软件生命周期质量保证与测试

目 录

CONTENTS

软件生命周期质量保证与测试

目 录

CONTENTS

软件生命周期质量保证与测试

第1章

软件质量概论

1.1 概述

信息技术的飞速发展，使软件产品应用到社会的各个领域，软件产品的质量自然成为人们共同关注的焦点。不论软件的生产者还是软件的使用者，均生存在竞争的环境中。软件开发商为了占有市场，必须把产品质量作为企业的重要目标之一，以免在激烈的竞争中被淘汰出局。用户为了保证自己业务的顺利完成，当然希望选用优质的软件。质量不佳的软件产品不仅会使开发商的维护费用和用户的使用成本大幅增加，还可能产生其他的责任风险，造成公司信誉下降，继而冲击股票市场。在一些关键的应用中，例如民航订票系统、银行结算系统、证券交易系统、自动飞行控制软件、军事防御和核电站安全控制系统等，若使用质量有问题的软件，还可能造成灾难性的后果。

软件危机曾经是软件界甚至整个计算机界最热门的话题，为此，无数软件从业人员和专家都付出了大量的努力想要解决这场危机。但是，随着对软件危机认识的深入，人们已经逐步意识到所谓的软件危机实际上仅是一种状况，那就是软件中有错误，正是这些错误导致了软件开发在成本、进度和质量上的失控。有错是软件的属性，而且是无法改变的，因为软件是由人来完成的，所有由人做的工作都不会是完美无缺的。问题在于如何去避免错误的产生和消除已经产生的错误，使程序中的错误密度达到尽可能低的程度。

1.2 软件生命周期概述

1.2.1 软件生命周期

软件生命周期是指软件的产生直到报废的生命周期，周期内有问题定义、可行性分析、总体描述、系统设计、编码、调试和测试、验收与运行、维护升级到废弃等阶段。一般来说，软件的整个生命周期可以分为如下 6 个阶段。

（1）问题的定义及规划

此阶段是软件开发方与需求方共同讨论，主要确定软件的开发目标及其可行性。

（2）需求分析

在确定软件开发可行的情况下，对软件需要实现的各个功能进行详细分析。需求分析阶段是一个很重要的阶段，这一阶段做得好，将为整个软件开发项目的成功打下良好的基础。“唯一不变的是变化本身”，同样需求也是在整个软件开发过程中不断变化和深入的，因此我们必须制定需求变更计划来应付这种变化，以保证整个项目的顺利进行。

（3）软件设计

此阶段主要根据需求分析的结果，对整个软件系统进行设计，如系统框架设计、数据库设计等。软件设计一般分为总体设计和详细设计。好的软件设计将为软件程序编写打下良好的基础。

（4）程序编码

此阶段是将软件设计的结果转换成计算机可运行的程序代码。在程序编码中必须要制定统一并符合标准的编写规范，以保证程序的可读性、易维护性，提高程序的运行效率。

（5）软件测试

在软件设计完成后要经过严密的测试，以发现软件在整个设计过程中存在的问题并加以纠正。整个测试过程分单元测试、组装测试、确认测试、系统测试和验收测试五个阶段进行。测试的方法主要有白盒测试和黑盒测试两种。在测试过程中需要建立详细的测试计划并严格按照测试计划进行测试，以减少测试的随意性。

（6）运行维护

软件维护是软件生命周期中持续时间最长的阶段。在软件开发完成并投入使用后，由于多方面的原因，软件不能继续适应用户的要求，要延续软件的使用寿命，就必须对软件进行维护。软件的维护包括纠错性维护和改进性维护两个方面。

软件生命周期模型的发展实际上体现了软件工程理论的发展。在早期，软件的生命周期处于无序、混乱的状态。这种状态直接导致了软件危机的出现，人们开始寻找产生危机的内在原因，发现其原因可归纳为两方面。一方面是由软件生产本身的复杂性所导致，另一方面是由软件开发所使用的方法和技术的多样性导致。

为克服软件危机，人们研究和借鉴了工程学的某些原理和方法，并形成了一门新的学科，即软件工程学，但由于软件个性化要求非常突出，在快速变化的商业环境下，不断变更的需求可能是一种残酷的事实，而这些都会严重影响软件产品的质量。因此，我们有必要讨论如何通过测试提高软件产品的质量，尤其重要的是如何在早期介入测试并发现问题等。

1.2.2　软件开发过程模型

随着计算机应用的飞速发展，软件的复杂程度不断提高，源代码的规模越来越大，软件开发过程越来越不容易被控制。在长期的研究与实践中，人们越来越深刻地认识到，建立简明准确的表示模型是把握复杂系统的关键。为了更好地理解软件开发过程的特性，跟踪、控制和改进软件产品的开发过程，就必须对这一开发过程进行模型化处理。

模型是对事物的一种抽象，人们常常在正式建造实物之前，首先建立一个简化的模型，以便更透彻地了解它的本质，抓住问题的要害。在模型中，先要剔除那些与问题无关的、非本质的东西，从而使模型与真实的实体相比更加简单明了、易于把握。总的来说，使用模型可以使人们从全局上把握系统的全貌及其相关部件之间的关系，可以防止人们过早地陷入各个模块的细节。

经过软件领域的专家和学者的不断努力，从早期的瀑布模型到现在的极限编程模型、Rational 统一过程模型（RUP）、微软公司的 MSF 过程模型等，各种软件过程模型不断被推出。目前主要有瀑布模型、原型模型、快速应用开发（RAD）模型、螺旋模型、增量模型和迭代模型、构件组装模型、并发模型、测试驱动模型（TDD）的软件开发、Rational 统一过程模型和 UML、协议开发——形式描述技术 FDT、敏捷方法——极限编程模型等。模型来源于实践，又应用于实践，新兴的模型取代落后的模型，如著名的软件开发瀑布模型，简单清晰，代表了传统的软件工程思想，但现在看来，瀑布模型不符合软件开发的实际情况，容易将开发引入误区。因为开发的许多工作是并行的，如 V 模型所揭示的，绝不是简单的顺序过程。而且每个阶段不是不能逆转的，如原型模型所揭示的那样多次往返过程。相反，认识到在软件开发过程中，克服与用户沟通困难性，强调人的重要性，需要不断提高开发效率，缩短开发周期及产品后期的维护性，如原型模型、V 模型和敏捷方法等越来越受到重视。

1．瀑布模型

瀑布模型规定了各项软件工程活动，包括制定开发计划、进行需求分析和说

明、软件设计、程序编码、测试及运行维护，如图 1-1 所示。图 1-1 显示出自上而下、相互衔接的固定次序，如同瀑布流水，逐级下落。

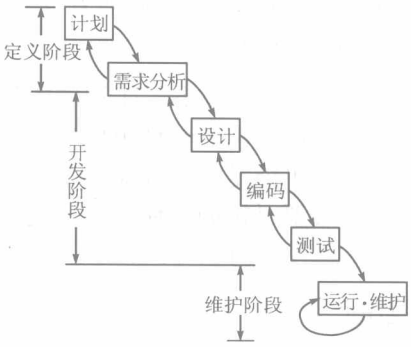

图 1-1 软件生命周期的瀑布模型

然而软件开发的实践表明，图 1-1 中各项活动之间并非完全是自上而下，呈线性图式。实际情况是每项开发活动均应具有以下特性：

（1）从上一项活动接受该项活动的工作对象，作为输入；

（2）利用这一输入实施该项活动应完成的内容；

（3）给出该项活动的工作成果，作为输出传给下一项活动；

（4）对该项活动实施的工作进行评审。若工作得到确认，则继续进行下一项活动，在图 1-1 中用向下指的箭头表示；否则返回前项，甚至更前项的活动进行返工，在图 1-1 中用向上指的箭头表示。

需要注意：软件维护在软件生命周期中有它的特点。一方面，维护的具体要求是在软件投入运行以后提出来的，经过“评价”，确定变更的必要性，才进入维护工作。另一方面，维护中对软件的变更仍然要经历上述软件生命周期在开发中已经历过的各项活动。如果把这些活动一并表达，就构成了生命周期循环，如图 1-2 所示。事实上，有人把维护称为软件的二次开发，正是出于这种考虑。由于软件在投入使用以后可能经历多次变更，为把开发活动和维护活动区别开来，便有了如“b”形的软件生命周期表示，如图 1-3 所示。它与前述的软件生命周期循环一样，都是软件生命周期瀑布模型的变种。

瀑布模型为软件开发和软件维护提供了一种有效的管理图式。根据图 1-3 制定开发计划，进行成本预算，组织开发力量，以项目的阶段评审和文档控制为手段有效地对整个开发过程进行指导，从而保证了软件产品及时交付，并达到预期的质量要求。瀑布模型 20 多年来之所以广为流行，是因为它在消除非结构化软件、降低软件的复杂度、促进软件开发工程化方面起着显著作用。与此同时，瀑布模型在大量的软件开发实践中也逐渐暴露出它的严重缺点。其中最为突出的缺点是

该模型缺乏灵活性，特别是无法解决软件需求不明确或不准确的问题。这些问题的存在对软件开发会带来严重影响，最终可能导致开发出的软件并不是用户真正需要的软件，并且这一点在开发过程完成后才有所察觉。面对这些情况，无疑需要进行返工或是不得不在维护中纠正需求的偏差。但无论上述哪一种情况都必须付出高额的代价，并将为软件开发带来不必要的损失。另一方面，随着软件开发项目规模的日益庞大，由于瀑布模型不够灵活等缺点引发出的上述问题显得更为严重。为弥补瀑布模型的不足，近年来已经提出了多种其他模型。

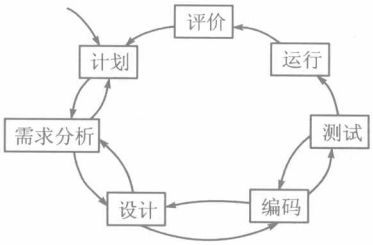

图 1-2　软件生命周期循环

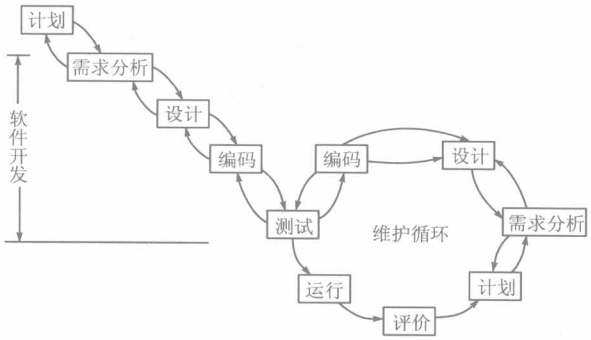

图 1-3　具有维护循环的软件生命周期

2．演化模型

由于在项目开发的初始阶段人们对软件的需求认识常常不够清晰，因而使得开发项目难以做到一次开发成功，出现返工再开发在所难免。有人说，往往要“开发两次”后开发出的软件才能较好地令用户满意。第一次是试验开发，其目标只是在于探索可行性，弄清软件需求；第二次则在此基础上获得较为满意的软件产品。通常把第一次得到的试验性产品称为“原型”。显然，演化模型在克服瀑布模型缺点、减少由于软件需求不明确而给开发工作带来风险方面，有显著的效果。

3．螺旋模型

对于复杂的大型软件，开发一个原型往往达不到要求。螺旋模型将瀑布模型与演化模型结合起来，并且加入两种模型均忽略了的风险分析，弥补了两者的不足。

在此先简要说明什么是风险分析。“软件风险”是普遍存在于任何软件开发项目中的实际问题。对不同的项目，其差别只是风险有大有小而已。在制定软件开发计划时，系统分析员必须回答，项目的需求是什么，需要投入多少资源以及如何安排开发进度等一系列问题。然而，若要他们当即给出准确无误的回答是不容易的，甚至几乎是不可能的。但系统分析员又不可能完全回避这一问题。凭借经验的估计给出初步的设想难免带来一定的风险。实践表明，项目规模越大，问题越复杂，资源、成本、进度等因素的不确定性越大，承担项目所冒的风险也越大。总之，风险是软件开发不可忽视的潜在不利因素，它可能在不同程度上损害

到软件开发过程或软件产品的质量。软件风险驾驭的目的是在造成危害之前，及时对风险进行识别、分析、采取对策，进而消除或减少风险的损害。

螺旋模型沿着螺线旋转，在笛卡儿坐标的 4 个象限上分别表达了 4 个方面的活动，如图 1-4 所示。

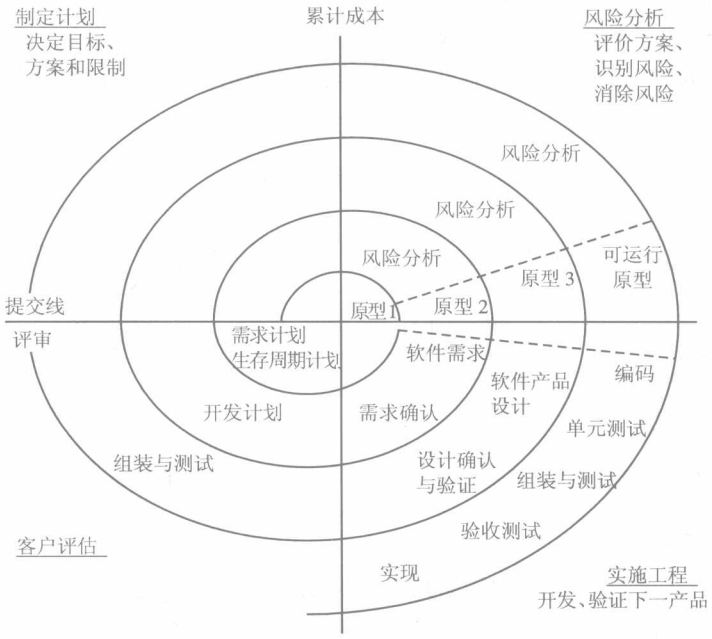

图 1-4 螺旋模型

（1）制定计划——确定软件目标，选定实施方案，弄清项目开发的限制条件；

（2）风险分析——分析所选方案，考虑如何识别和消除风险；

（3）实施工程——实施软件开发；

（4）客户评估——评价开发工作，提出修改意见。

沿螺线自内向外每旋转一圈便开发出更为完善的一个新的软件版本。例如，在第一圈，确定了初步的目标、方案和限制条件以后，转入右上象限，对风险进行识别和分析。如果风险分析表明，需求有不确定性，那么在右下的工程象限内，所建的原型会帮助开发人员和客户，考虑其他开发模型，并对需求做进一步修正。

客户对工程成果做出评价之后，给出修正建议。在此基础上需再次计划，并进行风险分析。在每一圈螺线上，风险分析的终点是判断是否继续下去的凭证。假如风险过大，开发者和用户无法承受，项目有可能终止。多数情况下，沿螺线的活动会继续下去，自内向外，逐步延伸，最终得到期望的系统，图 1-5 是螺旋