

21世纪软件开发、测试、维护和软件质量管理、验收必备用书

软件工程 与软件测试自动化 教程

张克东 庄燕滨 编著
熊继光 常明华 主审



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

<http://www.phei.com.cn>

21 世纪软件开发、测试、维护
和软件质量管理、验收必备用书

软件工程与软件测试自动化教程

张克东 庄燕滨 编著
熊继光 常明华 主审

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

测试是软件开发的一个重要环节。本书全面、系统地论述软件工程与软件测试自动化的理论与技术。全书共分 16 章,前 4 章简要地回顾与介绍软件工程及软件工具和软件测试理论;从第 5 章开始到第 16 章,结合 Panorama 着重对当今国际上最新的软件自动化再工程和测试技术进行全面、系统地阐述。内容包括软件危机和软件工程、软件工具、软件测试的一般性理论、软件测试用例设计、软件测试难点与 ISA 的相关解决、软件自动化测试质量的规范和度量、Panorama 测试软件的安装和设置、软件系统结构与系统流程的自动分析、软件质量自动评估分析、软件测试文档的自动生成、软件自动测试和测试用例生成、内存自动检查分析、故障自动跟踪分析、软件测试用例的自动回收、软件自动化测试的实例分析等。

在所附光盘中给出了全部 Panorama 测试软件产品及有关说明,供读者参考。

本书理论联系实际,反映当今世界软件测试的最新技术成果,是目前难得的好书。

本书可以作为大专院校计算机软件和计算机技术专业的教材,也可作为从事计算机软件开发的科技人员和软件项目高级管理人员学习工程化和软件测试自动化技术的参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有,侵权必究。

图书在版编目(CIP)数据

软件工程与软件测试自动化教程/张克东,庄燕滨编著. —北京:电子工业出版社, 2002.2

ISBN 7-5053-7307-2

I. 软… II. ①张… ②庄… III. ①软件工程—自动化—教材②软件—测试—自动化—教材 IV. TP311.56

中国版本图书馆 CIP 数据核字(2001)第 089756 号

责任编辑:龚兰方

印 刷:北京兴华印刷厂

出版发行:电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销:各地新华书店

开 本:787×1092 1/16 印张:17.5 字数:448 千字 附光盘 1 张

版 次:2002 年 2 月第 1 版 2002 年 2 月第 1 次印刷

印 数:5 000 册 定价:32.00 元(含光盘)

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。
联系电话:(010)68279077

序一

当前,信息化浪潮席卷全球,信息化程度已成为国力的重要标志,引起世界各国的高度重视。我国第九届全国人民代表大会第四次会议通过的“中华人民共和国国民经济和社会发展第十个五年计划纲要”中明确把“大力推进信息化”、“以信息化带动工业化”作为国家方针,并采取相应措施,这无疑对我国的发展是非常重要的。

但是,在信息化过程中,应用信息技术解决实际问题并带来效益(包括经济效益和社会效益)是根本。这实际上是一系列的应用信息系统的开发、运行和改进的过程,而要达到此目的,除了认识、机制、体制、政策及人才等因素外,就技术层面上看,软件(包括系统软件,应用软件及中间件等),尤其应用软件是非常关键的一环,它是应用系统的灵魂,体现应用系统的实质与特征。因此,软件的质量与可靠性将给应用系统带来极大的影响,在一定程度上决定着应用系统的命运。

许多实践说明,由于软件的问题造成应用系统不能正常运转或停顿的占相当比例(其他原因则是硬件、网络和操作等),在我国许多部门的系统中都出现过类似情况,它造成了一定损失,这些软件的开发往往没有严格而规范的管理和测试。本人曾在 20 世纪 60 年代到 20 世纪 70 年代初参加过有一定规模的信息系统工程项目工作,当时的软件开发,虽有整体设计、明确的各功能模块间接口关系,且经过单元测试、联调和系统综合测试,但在实际应用中还是问题百出,最后得经过相当长时间的试验与修改才能达到要求,而这是要花出相当的代价的。后来在总结中提出了“软件可靠性”问题。

应该说,要绝对保证软件的质量与可靠性是很困难的:这除了相当长时间以来软件开发是带有“个人智慧与技巧”的性质;同时也应该看到软件要去满足千差万别和千变万化的越来越复杂的应用需求,使得软件功能烦多,复杂而庞大;再加上开发方式和质量保证的技术与方法又滞后于需求的发展。

长期以来,软件开发的效率、质量与可靠性保证,在某种程度上已制约着应用系统的发展。因此,许多年以来,人们都非常关注并进行了大量工作。20 世纪 70 年代提出软件工程方法,强调生命周期、结构化分析与设计技术;20 世纪 80 年代提出软件开发管理(推出 SW - CMM),又有 ISO - 9000 标准,软件质量保证(SQA)等等。与此同时,人们在软件开发的自动化和相关工具的研制上也开展了许多工作并取得了不少成果,我们国家也独立地开展了许多这方面的工作。

关于上述成果情况,国内有不少书籍和文章介绍,尤其在软件工程方面较多,但在关于软件自动化测试方面的相对较少。因此,当看到这本由张克东和庄燕滨先生编著的《软件工程和软件测试自动化教程》一书时,非常高兴。通过此书可了解到当前国际上软件自动测试技术的现状与发展,这对我国正在发展着的软件产业会有很大的好处,将会对从事软件研制的管理者、科技人员和应用开发人员有所帮助。

国家信息中心 李正男

序二

软件测试是软件开发的一个重要环节。它的重要性不仅表现在软件测试工作上而且直接关系到软件产品的质量,它需要具有一定水平和实践能力的人员参与;而且软件项目的测试成本要占整个软件产品开发成本的一半,甚至更多。没有哪一个软件开发项目和哪一个软件开发组织不开展测试工作的。而如何把这样一项重要的工作做好,却是非常值得我们关注的问题。

有些软件项目的负责人在谈论自己项目的测试工作时,似乎以投入了多少测试工作量和使用了多少测试用例来表明如何重视测试工作。他们却恰恰忽略了测试的效率和测试的效果问题。比如,在测试完成后,他们往往不能回答测试的充分性如何,测试的有效性如何。对于这一点,也许我们可以用公安人员从事破案工作与软件测试做一类比。十分明显,破案工作并不能以投入了多少警力来评价其成绩,而应考察公安机构的破案率。

在此需要指出的是,要想取得测试充分性数据,没有相应的工具是难以做到的。其实,不仅测试工作,就软件开发项目总体来说,都需要有软件工具的支持。特别是当前,我国软件开发组织迫切需要得到各种软件工具,用以切实提高软件开发效率,进而改变软件开发的所谓“手工作坊式”生产的落后面貌。

本书作者着重介绍了旅美华人学者熊继光先生组织开发的优秀软件测试工具 Panorama。这套工具在国际上已获得了知名用户的广泛欢迎,享有很高的声誉。书中材料编写得系统、详实、易于理解。我相信,本书的出版以及这一测试工具的引进将使在第一线开发的软件人员受益。他们会感到,测试工作不再是低效、烦琐的事,同时他们也会从中受到启发,提高在软件开发中自觉地采用工具的意识,最终促进我国软件产业的发展,实现熊先生和作者的报国愿望。

清华大学 郑人杰

序三

几十年以前,在计算机信息处理系统的总费用中,大约只有百分之十五花费在软件上。然而,今天的情况正好相反,即使这样,软件开发工程方面还存在不少的问题和难点:

- 相比较而言,软件开发技术的发展较硬件开发技术的发展要慢得多,许多软件工程的业务至今仍然靠手工来完成。

- 随着立体图像处理 and 多媒体等新技术的出现,今天的许多软件系统已经变得十分庞大而复杂。

- 由于至今软件依然采用文本文件格式(非图形化)来书写,所以它们是难于阅读 and 理解的。

- 由于无法从理论上 and 实践上证明一个复杂的软件系统本身是否百分之百正确,所以软件开发者必须通过大量的测试来试图找出软件源程序中的错误。一旦软件经过修改后,还得对该软件重新进行测试——这就是为什么许多软件公司的开发费用的一半左右都花费在测试上。

- 大量实际测试结果表明,如果单独采用功能测试(Black - Box)的方法,则在那“完全测试通过”的软件中,通常还会有一半左右的程序代码从未被执行过。结构测试(White-Box)方法可以被用来提高程序代码的测试覆盖率,但结构测试方法如果没有自动化的工具是非常难以进行的——首先碰到的问题就是,怎么知道哪个程序分支是否被执行过?

- 软件系统时常修改、补充以满足新的需求 or 添加新的特性。然而,软件的修改是非常费时费力的——因为对某个模块的小小修改可能牵涉到别的许多模块都要作相应的修改。

- 因为软件设计文档常常不完整,不能在线访问,或者在软件代码修改后还来不及修改与程序源代码不一致等,使得软件的维护十分困难。

美国国际软件自动化(ISA)公司的 Panorama For C/C + + , Java 和 VB 产品就是被设计来解决今天软件系统所面临这些问题的。它给软件开发生存周期的许多方面带来了自动化,包括软件结构分析与逻辑框图作图的自动化,软件静态分析、数据分析、复杂性分析与分析结果列表的自动化,软件质量分析、动态性能分析、软件代码分支或条件测试覆盖率分析、软件测试用例有效性分析与测试用例最小集的自动选取以及软件界面手工操作过程的自动记录与自动再执行(Playback)等。它的有效利用能使软件系统变得容易理解、测试、评价、修改和维护。在美国权威人士 Edward Kit 的“Software testing in the real world”(1995)一书中所作全部 5 大类指标的评价里,ISA 公司的 Panorama 产品有三项被列入第一 or 第二。

ISA 公司的 Panorama 系列软件工程和测试自动化产品已经为世界上许多著名的大公司所采用,如: SUN, HP, SIEMENS, SONY, TOSHIBA, FUJITSU, XEROR, CANON, TOKYO, FANAC, IBM 及 MICROSOFT 等。

常州爱夏软件自动化应用有限公司的张克东先生 and 庄燕滨先生为介绍 and 反映当今国际

上软件自动化测试技术方面的许多新成果和发展动向而编写的此份教材,对于促进国内软件的产业化和培养高素质的软件技术人才,将会是十分有益的。

美国国际软件自动化(ISA)公司总裁
熊继光

序四

人类自开始使用劳动工具时,便逐步将自己区别于其他动物。人类劳动工具的逐步发展,标志着人类社会的进步。人类劳动工具的先进性是一个社会的生产力先进程度的重要标志。在现代社会中,劳动工具先进与否,实际上存在一个全球性的、客观的检测和评价的标准和方法。

计算机和计算机软件都是人类劳动的成果,也是人类劳动的工具。计算机硬件的制造和检测早已形成国际共识,有了国际公认的质量标准。计算机软件产业正在蓬勃发展,但在计算机软件开发工具的先进性、准确性的检测方面在全球范围内还存在较大的不平衡性。就我们身边而言,软件开发“手工作坊式”的状况还没有彻底改变。软件测试的重要性已引起越来越多的人的重视,但还未形成社会规范。特别是还缺少掌握先进软件测试工具和方法及理念的专业测试人才。

高等院校的主要任务是培养国家经济建设和社会发展急需的专门人才,另一重要任务是推进科技创新和科技向现实生产力的转化。常州工学院和常州爱夏软件自动化应用有限公司致力于彻底改变我们周围“手工作坊式”的软件开发环境,致力于为国家培养大批具有“团队精神”的高层软件开发人才,培养大量从事“软件质量保障”的软件测试工程师。张克东先生和庄燕滨先生投身于这项事业并专门编写了这本教材。这对我国软件开发和软件测试人才的培养,以及软件业的发展和规范,无疑是一件十分有益,极具先导性的事情。希望这本书问世以后,很快成为高水平软件开发人才和软件测试人才的良师益友。

常州工学院

院长

爱夏软件自动化应用有限公司

董事长

马树杉

前 言

软件工程的主要任务是通过提供规范的分析设计方法及软件工具,来避免或减少软件错误的发生,为最终根除软件危机提供强有力的技术保障。而软件测试的目的是发现故障,测试至少要耗费运行程序工作量的一半。显然,熟悉测试是在软件开发周期中节省成本和减少时间的主要方法。同时,软件测试是对创造力和智力非常有挑战性的任务。测试一个大型软件需要的智能要超过设计这个程序的智能。软件在它发行之前应当通过彻底的测试,以保证它的可靠性和功能性,不幸的是,测试工程师要覆盖一个大型程序的所有情况会感到太麻烦和费时。确实,软件的每个部分如能被分别测试,同时一些指定的路径也能被测试,这样,在总的软件质量上有有效的改进也就可以达到。平均来说,如没有测试覆盖分析工具,软件在发行前仅有 50%的源程序被测试过。在差不多有一半源代码没有被测试的情况下,大量的故障(bug)随软件一道被发行出去。在这种情况下,软件的质量、性能和功能不可能得到保障。Panorama 在许多方面提供了全面的解决,并且在某些方面,对软件测试的贡献是独一无二的。

本书共分 16 章:前面 4 章简要地回顾与介绍一般的软件工程及软件工具和软件测试理论;从第 5 章开始到第 16 章着重结合 Panorama 对软件自动化再工程和测试当今国际上最新的技术进行比较全面系统的阐述。

本书第 1,2,3,4,5 章和第 16 章由庄燕滨编写;其余章节及附录由张克东编写,并由张克东统编全书。

2000 年国家高技术研究发展计划(863 计划)已经将软件质量保障平台(软件质量保障集成平台和软件质量保障集成平台的关键技术 ZD1202)列入信息技术领域中的重大项目。为了发展国家软件事业和促进软件产业化,培养高素质的软件技术人才,我们特编此教材,略尽微薄之力。

本书在编写过程中得到美国 ISA 公司和常州工学院计算机系的大力支持和合作。在此要特别感谢常州爱夏软件自动化应用公司董事长、常州工学院马树杉院长,国家信息中心原副主任兼总工李正男高级工程师,著名的国内软件工程和软件测试学术界权威、清华大学郑人杰教授,以及美国 ISA 公司总裁、杰出的华人实业家、软件专家熊继光先生分别热情地为本书作序。

本书由 ISA 公司总裁熊继光、常州工学院计算机系主任常明华负责主审,ISA(上海)公司陈力群、赵昌明审阅了本书的主要内容,此外,杨建军(博士生)、姚砺(博士生)、张溟、石敏等在校核、整理等方面作了大量的努力;常州爱夏软件自动化应用有限公司的汤文君及徐琴芳在编排、绘图、录入等工作中,也都付出了辛勤的劳动,使本书得以尽快地和读者见面。在此谨向以上各位表示衷心感谢。

为方便广大读者学习和掌握 Panorama 测试软件,我们将全部 Panorama 测试软件产品及

有关使用说明刻录成光盘(CD - ROM)随本书一起发行。

本书虽经反复推敲,试图概括和着重反映当今国际上软件自动化测试技术的最新成果和发展,但限于作者水平,仍难免会有错误和不当之处,恳请读者批评指正,有什么意见和要求请与我们联系(mail to: zhangkd@pub.cz.jsinfo.net)。

编 著 者

目 录

第1章 软件危机与软件工程	(1)
1.1 软件危机	(1)
1.1.1 什么是软件危机	(1)
1.1.2 软件危机内在的原因	(1)
1.1.3 软件工程与软件危机的解决	(1)
1.2 软件质量与可靠性	(2)
1.2.1 软件质量因素和质量特性	(3)
1.2.2 软件可靠性	(4)
1.3 软件工程方法论——CMM 模型	(5)
1.3.1 SQA 计划	(6)
1.3.2 ISO 9000 质量保证体系	(6)
1.3.3 CMM 模型	(7)
第2章 软件工具和面向对象软件工程	(9)
2.1 传统软件工程的困境及解决	(9)
2.1.1 传统软件工程面临的困境浅析	(9)
2.1.2 软件工具和再工程环境	(10)
2.1.3 面向对象的软件工程	(11)
2.2 经典的软件工程与建模工具	(11)
2.2.1 业务调查阶段	(11)
2.2.2 需求定义阶段	(14)
2.2.3 总体设计阶段	(15)
2.2.4 详细设计阶段	(16)
2.2.5 程序生成阶段	(17)
2.3 Panorama 自动化测试工具和支撑环境	(17)
2.3.1 系统特点	(17)
2.3.2 采用领先的技术	(18)
2.3.3 Panorama 工具系列和功能	(19)
第3章 软件工程范畴中软件测试的一般性理论	(23)
3.1 基本概念	(23)
3.1.1 软件测试基础	(23)
3.1.2 静态方法和动态方法	(25)
3.1.3 黑盒测试和白盒测试	(26)
3.1.4 软件测试的步骤	(28)
3.1.5 测试阶段的信息流	(29)
3.2 单元测试	(29)
3.2.1 单元测试的考虑	(30)
3.2.2 单元测试的过程	(32)

3.3 集成测试	(32)
3.3.1 非增式测试	(33)
3.3.2 增式测试	(33)
3.3.3 不同集成测试策略的比较	(35)
3.4 确认测试	(35)
3.4.1 确认测试准则	(36)
3.4.2 配置审查	(36)
3.5 系统测试	(36)
3.5.1 恢复测试	(36)
3.5.2 安全测试	(37)
3.5.3 强度测试	(37)
3.5.4 性能测试	(37)
3.6 验收测试	(37)
3.6.1 验收测试的范围	(37)
3.6.2 软件配置及文档资料	(38)
3.7 调试	(38)
3.8 面向对象的软件测试	(39)
3.8.1 在 OO 语境中的单元测试	(39)
3.8.2 在 OO 语境中的集成测试	(39)
3.8.3 在 OO 语境中的确认测试	(40)
第 4 章 软件测试用例的设计	(41)
4.1 黑盒测试法和测试用例的设计	(41)
4.1.1 等价类划分	(41)
4.1.2 边界值分析	(42)
4.1.3 因果图	(43)
4.1.4 比较测试	(43)
4.2 白盒测试法和测试用例的设计	(43)
4.2.1 基本概念	(44)
4.2.2 逻辑覆盖	(47)
4.2.3 路径分析	(53)
4.3 针对专门环境和应用的测试	(59)
4.3.1 GUI 测试	(59)
4.3.2 客户/服务器体系结构的测试	(60)
4.3.3 实时系统的测试	(60)
第 5 章 软件工程及测试的难点和 ISA 的相关解决	(62)
5.1 新系统开发支持	(63)
5.1.1 支持新系统设计	(63)
5.1.2 减少故障和编码风险	(64)
5.1.3 帮助软件工程师解决调试问题	(65)
5.1.4 源代码级程序终止和实时运行错误定位	(66)
5.2 为已有软件系统提供修改设计、测试和质量保障支持	(67)
5.2.1 有效率的模块级源文件编辑和安全的代码修改	(68)

5.2.2	帮助代码理解、检查和通过	(68)
5.2.3	为单元和系统集成测试增加有序性	(69)
5.2.4	基于多重度量的复杂性分析的测试计划	(71)
5.2.5	测试覆盖分析与工程文件(.mak)驱动代码插装	(72)
5.2.6	有效的未测试路径(或段)的测试和再测试	(73)
5.2.7	节省时间的模块级再测试	(74)
5.2.8	有效的花费和自动的系统级再测试	(75)
5.2.9	面向对象的软件质量度量的测量	(76)
5.2.10	在 Windows 上基于 GUI 的复原测试	(77)
5.3	再工程支持	(78)
5.3.1	系统结构分析	(78)
5.3.2	系统的类/过程分析	(79)
5.3.3	系统过程的逻辑分析和流程图化	(80)
5.3.4	系统控制流和 Cyclomatic 度量复杂性分析	(81)
5.3.5	系统数据分析	(82)
5.3.6	系统性能分析	(83)
5.3.7	可联机访问的文档手册和报告自动生成	(84)
5.4	其他	(85)
5.4.1	工程管理	(85)
5.4.2	训练新成员	(86)
5.4.3	定制系统说明和验收评估	(87)
第 6 章	软件自动化测试的质量规范和度量	(88)
6.1	Panorama 自动化测试软件质量保障体系特点	(88)
6.1.1	Panorama 测试覆盖的特点	(88)
6.1.2	测试用例和代码的对应分析	(89)
6.1.3	测试用例的最小化	(90)
6.1.4	意外结束的定位	(90)
6.1.5	复杂性分析	(90)
6.1.6	测试结果和程序逻辑的图形表达	(90)
6.1.7	用户友好的界面	(91)
6.2	Panorama 自动测试软件提供的质量保障体系	(91)
6.2.1	软件质量保障度量	(91)
6.2.2	Panorama 软件质量保障度量的含义	(92)
6.2.3	支持 IEEE 标准	(94)
第 7 章	Panorama 测试软件的安装和设置	(97)
7.1	Panorama C/C++ 的安装和设置(Windows NT/98 平台)	(97)
7.1.1	系统要求	(97)
7.1.2	如何安装	(97)
7.1.3	获得 Panorama 授权许可	(97)
7.2	Panorama C/C++ 系统设置及基本操作	(99)
7.2.1	主菜单窗口	(99)
7.2.2	工具栏	(100)

7.2.3	加载源程序(.mak)	(101)
7.2.4	创建一个分析数据库	(102)
7.2.5	生成动态数据	(104)
7.2.6	测试覆盖	(105)
7.3	Panorama for Java 和 JavaAnalyzer 的安装和设置	(105)
7.3.1	要求	(105)
7.3.2	如何安装	(105)
7.3.3	设置	(107)
7.4	Panorama VB 的安装和设置	(108)
7.4.1	系统要求	(108)
7.4.2	安装	(108)
7.4.3	启动 Panorama VB 及选定目标源程序	(108)
7.4.4	创建分析数据库	(109)
第 8 章	软件系统结构的自动分析(OO-Browser)	(111)
8.1	基本概念	(111)
8.1.1	函数调用图	(111)
8.1.2	类图	(111)
8.2	软件系统结构的自动测试分析技术	(112)
8.3	生成软件系统结构图	(113)
8.3.1	激活 Panorama OO-Browser	(113)
8.3.2	软件系统中类和函数的结构图	(113)
8.3.3	结构图菜单	(114)
8.4	软件系统中的子系统结构(子树)	(116)
8.5	函数和类的定位	(117)
8.6	测试数据的自动覆盖	(119)
8.6.1	显示编码次序	(119)
8.6.2	显示函数大小图	(119)
8.6.3	显示函数大小和复杂性图	(119)
8.6.4	显示测试覆盖结果	(120)
8.7	自动显示软件系统结构总体测试信息	(120)
8.7.1	显示有关结构图的信息	(120)
8.7.2	显示各个函数和类的测试信息	(120)
8.8	为选择的软件系统模块生成各种流程图	(122)
8.8.1	为选择的模块生成 J-Diagram 逻辑流程图	(122)
8.8.2	为选择的模块生成 ActionPlus 流程图	(123)
8.8.3	为选择的模块生成 J-Flow 控制流程图	(123)
8.9	改变观察系统结构图的方法	(123)
8.10	在系统结构图中编辑(修改)软件模块源代码	(125)
8.11	打印软件系统结构图	(126)
第 9 章	系统流程自动分析(OO-Diagrammer)	(129)
9.1	基本概念	(129)
9.2	Panorama 系统流程自动测试的技术要点	(129)

9.3	Panorama J-Diagrammer 流程图分析工具	(130)
9.3.1	函数高层逻辑图	(132)
9.3.2	类高层逻辑图	(132)
9.3.3	J-Diagram 逻辑流程图	(133)
9.3.4	J-Flow 控制流程图	(133)
9.3.5	ActionPlus 增强的代码流程图	(134)
9.3.6	OO-Diagrammer 自动提供测试信息汇总	(134)
9.4	围绕系统流程图的动态交叉引用和跳转	(136)
9.4.1	动态交叉引用	(137)
9.4.2	从交叉引用返回	(137)
9.5	利用标签保存观察窗	(138)
9.6	生成逻辑路径	(138)
9.7	显示路径信息	(138)
9.8	在流程图中定位代码	(139)
9.9	打印流程图	(141)
第 10 章	软件质量自动评估分析(OO-SQA)	(143)
10.1	软件质量度量的基本概念	(143)
10.1.1	软件质量是软件属性各种标准度量的组合	(143)
10.1.2	Panorama OO-SQA 的特点	(143)
10.1.3	Panorama OO-SQA 显示形式	(144)
10.1.4	Panorama OO-SQA 提供的软件度量类型	(144)
10.2	软件度量自动化的操作	(145)
10.2.1	激活 Panorama OO-SQA	(145)
10.2.2	生成软件度量	(145)
10.2.3	读软件度量	(146)
10.3	选择函数或类	(151)
10.3.1	选择函数	(151)
10.3.2	选定类	(152)
10.4	设置软件质量度量的期望值	(152)
10.5	打印度量图	(153)
10.5.1	在 PS 打印机上打印	(154)
10.5.2	输出软件度量图到普通打印机	(154)
第 11 章	软件测试文档的自动生成(OO-Analyzer)	(155)
11.1	基本概念	(155)
11.2	生成报告	(155)
11.3	阅读报告	(157)
11.4	在报告中搜索文本内容	(158)
11.5	保存一个报告	(159)
11.6	设置标签	(159)
11.6.1	附加标签	(159)
11.6.2	跳转到用标签标识的一个位置	(160)
11.6.3	清除标签	(160)

11.7 打印报告	(160)
第 12 章 软件自动测试和测试用例生成(OO-Test)	(162)
12.1 激活 Panorama/Test	(162)
12.2 测试用例的生成	(162)
12.2.1 为动态分析程序准备和运行测试用例	(162)
12.2.2 加载测试用例	(163)
12.2.3 创建测试用例	(164)
12.2.4 运行测试用例	(165)
12.3 Panorama 软件自动测试覆盖分析(TCA)	(166)
12.3.1 激活测试覆盖分析(TCA)	(166)
12.3.2 测试用例和文件/类/函数/段的对应	(167)
12.4 测试用例的效率	(169)
12.5 测试用例最小化	(171)
第 13 章 内存自动检查分析(OO-Memory)	(174)
13.1 基本概念	(174)
13.2 对于 C++ 内存检查工具设计的难点	(174)
13.3 操作	(174)
13.3.1 准备获得内存检查(Memory Checker)数据	(174)
13.3.2 激活 Panorama OO-Memory Checker	(175)
13.3.3 阅读 Memory Leak/Violation 报告	(176)
13.3.4 更新 Memory Leak/Violation 报告	(176)
13.3.5 其他	(177)
第 14 章 故障自动跟踪分析(OO-Defect Tracer)	(178)
14.1 基本概念	(178)
14.2 一个故障跟踪工具设计的难度	(178)
14.3 操作	(178)
14.3.1 准备得到故障跟踪数据	(178)
14.3.2 激活 Panorama OO-Defect Tracer	(178)
14.3.3 阅读 Defect Tracer(故障跟踪)报告	(179)
14.3.4 更新 Defect Tracer(故障跟踪)报告	(181)
14.3.5 其他	(181)
第 15 章 软件测试用例的自动回放(OO-Playback)	(182)
15.1 基本概念	(182)
15.1.1 为什么要俘获/回放	(182)
15.1.2 俘获/回放工具的分类	(182)
15.1.3 Panorama OO-Playback 新一代的俘获/回放工具	(182)
15.2 准备和运行测试用例	(183)
15.2.1 加载测试用例	(183)
15.2.2 创建测试用例	(184)
15.2.3 保存测试用例	(185)
15.2.4 运行测试用例	(186)
15.3 激活 Panorama/Playback	(186)

15.4 测试用例的自动回放	(187)
15.4.1 加载 .tdb 文件	(187)
15.4.2 回放测试用例	(188)
15.4.3 保存 playback 窗口回放的测试用例	(189)
15.4.4 打印回放窗口中回放的测试用例	(189)
第 16 章 软件自动化测试实例分析	(191)
16.1 一个测试覆盖自动化度量分析实例	(191)
16.1.1 示例程序	(191)
16.1.2 运行前的初始状态	(192)
16.1.3 第一组测试用例执行	(193)
16.1.4 第二组测试用例执行	(195)
16.1.5 第三组测试用例执行	(198)
16.1.6 第四组测试用例执行	(201)
16.1.7 第五组测试用例执行	(202)
16.1.8 第六组测试用例执行	(204)
16.2 一个稍大一点的 VB 程序自动化测试分析实例	(206)
16.2.1 利用 J-charts 得到系统的总体结构及覆盖信息	(206)
16.2.2 利用逻辑流程图详细理解代码逻辑	(210)
16.2.3 联机程序文档	(214)
附录 A 结构图的方框标识	(216)
附录 B 控制流程图(J-Flow)的标识	(217)
附录 C 逻辑流程图的标识	(221)
附录 D 在图上的宏表达	(225)
附录 E 词汇表	(226)
附录 F Panorama C/C++ For UNIX 版本的安装	(231)
附录 G 国际软件测试界权威人士的评价	(240)
附录 H 软件测试文件汇总	(243)
附录 I 软件企业如何实施基于 CMM 的过程改进	(255)
参考文献	(260)