

软件测试

Software Testing

兰雨晴 (北京航空航天大学软件工程研究所 100083)

摘要 电子政务系统建设过程中建立软件质量保证体系是必需和必要的。软件测试作为保证软件质量和可靠性的关键技术手段,是我们必须加以认真研究和对待的。本文着重介绍了软件测试过程中的关键软件测试活动,包括测试计划的编制、测试大纲编写、测试用例设计、测试执行过程控制、软件问题报告提交等内容,希望能对读者的电子政务工程实践有所帮助。

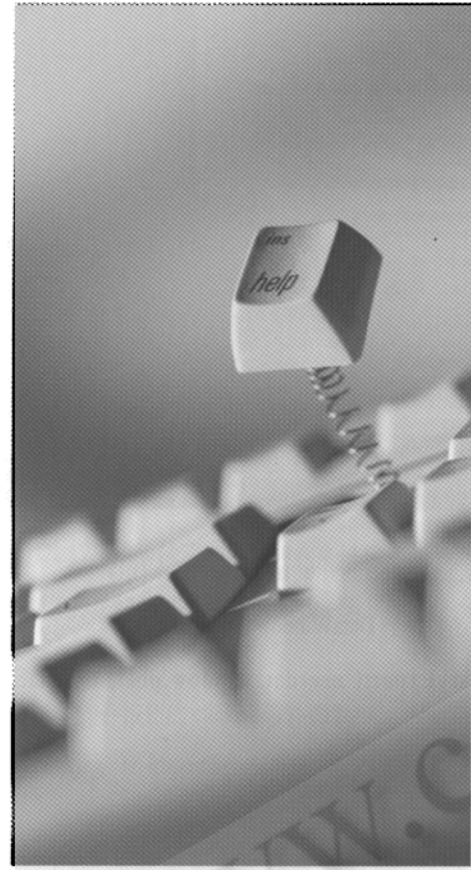
关键词: 电子政务 软件测试

软件测试是保证软件质量和可靠性的关键技术手段。在传统的软件开发瀑布模型中,软件测试被视为是紧随代码实现之后,揭示软件中的错误、保证和提高软件质量的重要阶段。在其他各种软件工程开发模型中,如快速原型、螺旋模型,以及面向对象的软件开发模型等,也都明确确认了软件测试的重要性和不可取代的地位。本文将在以下几节中简要介绍软件测试过程中测试实施过程的周期性,并阐明测试计划、测试大纲、测试用例、测试执行过程控制、软件问题

报告及其生命周期等基本概念,以及它们的编制与设计原则。

1 软件测试周期

在软件生存周期中,测试阶段又可分为几个子阶段,比如单元测试阶段、组合测试或集成测试阶段等。但是,不论在哪个测试阶段,其中的测试和改错工作总是反复交替地进行的。测试实施过程根据测试和改错活动之间的关系分为两种基本方式。



编者按:

伴随着我国信息化快速发展的势头,电子政务建设正式提上了日程,其必然的趋势和潜在的效益已不容置疑,而其浩大的声势和迅猛的来势也令人感到一些担忧,而这种担忧的一个关键就是建设中的软件质量控制,或更明确地讲,就是如何保证我们在各地和各部门即将开始的和正在建设的电子政务项目的成功,换言之,就是如何保证这些项目达到预期的建设目标,系统得以正常运行且发挥出预期的效益,从而提高办事效率,降低社会成本。同时,还能够最大限度地保证已建成项目的持续应用和发展。本专栏针对电子政务建设中软件质量控制所涉及的诸多方面,分四篇文章,分别对软件度量、软件测试、软件评审及相应的工具支持等进行了论述,以期对读者能有所帮助。

联系人: 兰雨晴 (北京航空航天大学软件工程研究所北京海淀区学院路37号 100083)

Lanyuqing@buaa.edu.cn 13910552180

010-82328213

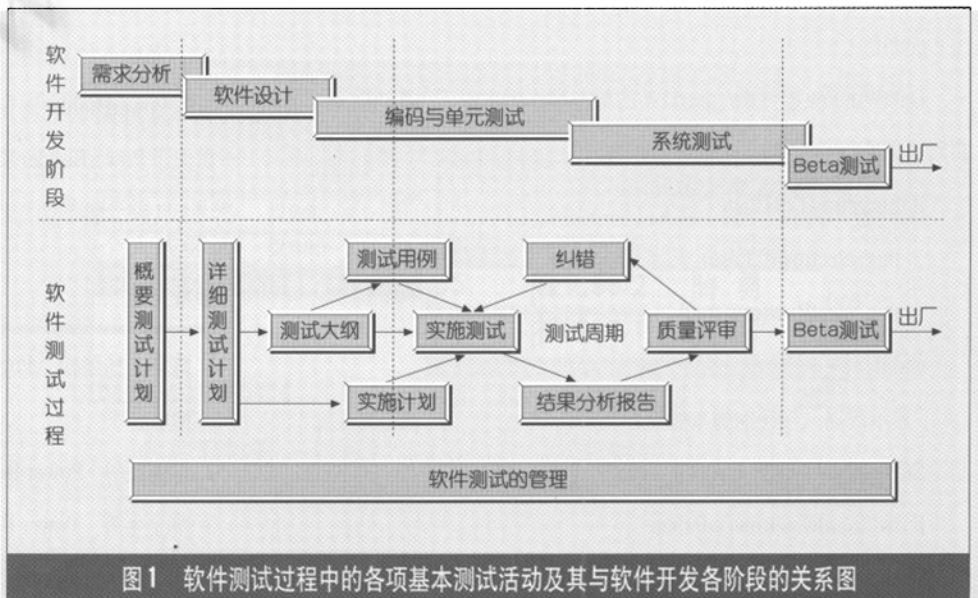


图1 软件测试过程中的各项基本测试活动及其与软件开发各阶段的关系图

(1) 串行方式: 测试和改错交替进行, 如图2所示。常见于单元测试阶段, 被测对象是一个独立的模块, 测试者是程序员本人:

(2) 并行方式: 测试与改错过程既是交错的, 同时又是并行进行的, 如图3所示。在集成测试阶段中, 测试一般应当由独立的软件测试人员来实施, 这种方法一方面可以有效地压缩测试的总周期, 但更重要的是可以避免开发者自身的思维局限, 更加客观全面地进行有效的测试。

在软件集成测试过程中, 测试的执行过程通常分为3个阶段(如图4所示):

(1) 初测期: 测试集中在系统的主要功能和关键的执行路径上。

(2) 细测期: 排除了主要障碍后, 测试进入细测期, 此时的测试将遍及系统大大小小的功能和方方面面的特性, 依据测试计划和测试大纲的要求, 逐一测试各项功能、性能、用户界面、兼容性、可用性等。这期间, 预期可以发现大量不同性质、不同严重程度的错误(Bugs)和问题(Questions)。

(3) 回归测试期: 系统已稳定, 一轮测试中发现的错误数目已十分有限, 这时便进入回归测试期, 此时的测试称作回归测试, 其主要目的是复查已知错误是否已经纠正, 并且确认纠错过程未引发任何新的错误。

最后一轮测试称作终结回归测试(Final Regression), 此时系统不再做任何修改, 此轮测试的目的在于对产品做最后确认, 以确保系统的质量已经达到预期的要求。

在集成测试过程中, 有两个重要的里程碑:

① 功能冻结(Function/Feature Freeze): 标志着整个系统的集成工作已经完成, 系统的功能、性能、界面、兼容性等均已经过测试, 符合设计要求, 确认系统功能和其他特性均不再做任何改变;

② 代码冻结(Code Freeze): 理论上, 系统



图2 单元测试阶段中典型的测试执行过程

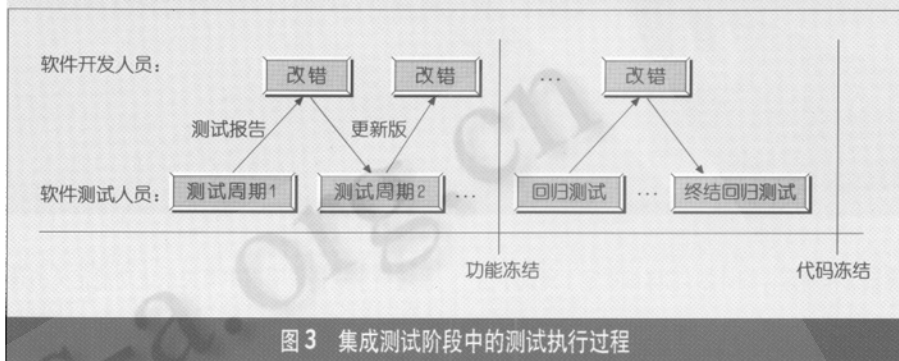


图3 集成测试阶段中的测试执行过程

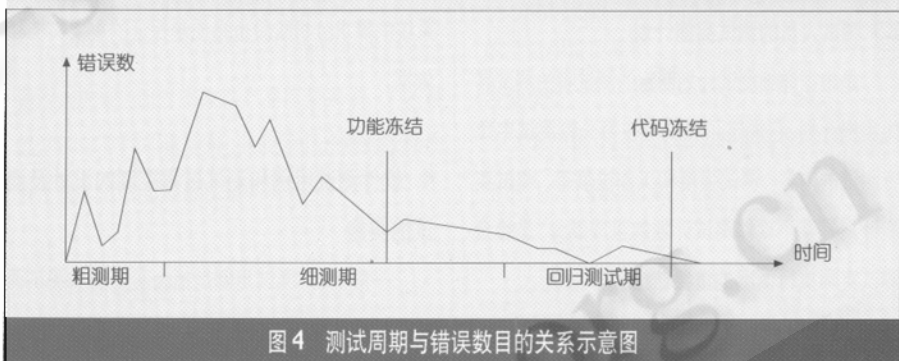


图4 测试周期与错误数目的关系示意图

需在达到无错误时方可冻结程序代码。但实际上, 由于人们无法证明系统是否无错, 甚至由于种种原因而不得不在保证系统质量的前提下保留下系统中少量不很严重的已知错误, 因此, 代码冻结只是标志着系统的当前版本的质量已经达到预期的要求, 冻结程序的源代码, 不再对其做任何修改。显然, 这个里程碑是设置在软件通过了终结回归测试之后。

2 软件测试计划

我们已经知道一个大中型软件系统的测试必须经历一个相当复杂的测试过程, 并且常常需要经历数月乃至更长时间, 需要投入相当可观的人力和物力资源, 所以针对整个项目的预定目标和可能的实际条件, 对系统的测试过程事先进行认真的计划, 并且在系统的开发和测试过程中, 根

据项目的实际进展情况和变化对测试计划及时进行调整和逐渐细化已成为软件测试过程中一项不可缺少的工作。

一般而言, 软件测试计划可分为3个层次。

2.1 概要测试计划

是软件项目实施计划中的一项重要内容, 应当在软件开发初期, 即需求分析阶段制定。这项计划应当定义被测对象和测试目标, 确定测试阶段和测试周期的划分; 制定测试人员、软硬件资源和测试进度等方面的计划; 规定软件测试方法、测试标准以及支持环境和测试工具, 比如, 语句覆盖率需达到95%、3级以上的错误改正率需95%、所有决定不改正的“轻微”错误都必须经过专门的质量评审委员会同意等。

2.2 详细测试计划

是针对子系统在特定的测试阶段所要进行的



测试工作制定详细计划,它详细规定了测试小组的各项测试任务、测试策略、任务分配和进度安排等。

2.3 测试人员的测试实施计划

是根据详细测试计划制定的测试者的测试具体实施计划,它规定了测试者在每一轮测试中负责测试的内容、测试强度和工作进度等。测试实施计划是整个软件测试计划的组成部分,是检查测试实际执行情况的重要依据。

3 软件测试大纲

软件测试大纲是软件测试的依据,它明确详细地规定了在一次测试中针对系统的每一项功能或特性所必须完成的基本测试项目和测试完成的标准。无论是自动测试还是手动测试,都必须满足测试大纲的要求。

实际上,测试大纲是从测试的角度对被测对象的功能和各种特性的细化和展开。比如,针对系统功能的测试大纲是基于软件质量保证人员对

系统需求规格说明书中有系统功能定义的理解,将其逐一细化展开后编制而成的。因此,测试大纲不仅是软件开发后期测试的依据,而且在系统的需求分析阶段也是质量保证的重要文档和依据。

4 软件测试用例和测试过程控制的设计及测试的实施

一般而言,测试用例是指为实施一次测试而向被测系统提供的输入数据、操作或各种环境设置。测试用例控制着软件测试的执行过程,因此,软件测试过程控制的设计与测试用例的设计也有着密切的关系。从广义上讲,软件测试执行过程的控制也是一种测试用例。测试用例的设计与生成是依据测试大纲对其中每个测试项目的进一步实例化。比如,对于一个输入项的测试,应当设计一组测试数据,包括合法的、边界的和非法的数据等。

从工程实践的角度讲,以下是几条设计测试

用例的基本准则。

4.1 测试用例的代表性

能够代表并覆盖各种合理的和不合理、合法的和非合法的、边界的和越界的,以及极限的输入数据、操作和环境设置等;

4.2 测试结果的可判定性

即测试执行结果的正确性是可判定的;

4.3 测试结果的再现性

即对同样的测试用例,系统的执行结果应当是相同的。

5 软件问题报告及其生命周期

软件问题报告是用来记录在测试过程中所发现的软件的问题。每一个软件问题报告都应包括:对问题的描述、可重现的步骤,问题所属的分类、严重性等主要元素。其基本的生命周期如图5所示:

笔者在过去的7年当中,一直在工程一线遵循上述测试过程参与完成了数十个软件测试项目,实践证明是成功的,因此也希望对读者有所帮助,特别是在电子政务建设中的软件质量保证有所帮助。 ■

参考文献

- 1 郑人杰、殷人昆、陶永雷,《实用软件工程》第二版,清华大学出版社,2001年4月版。
- 2 刘超、金茂忠,“软件测试过程的基本模型POCERM”,北航学报,1997/1,Vol.23, No.1, P56-60。

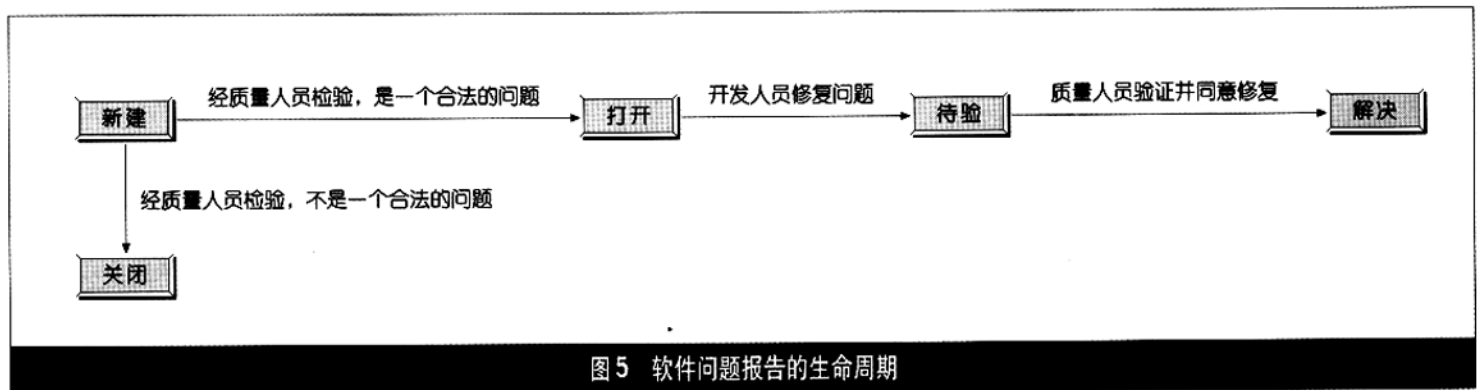


图5 软件问题报告的生命周期