

软件测试 Software Testing

20. 软件测试自动化

程适

cheng@snnu.edu.cn

计算机科学学院

2016 年 12 月 2 日



陕西师范大学
SHAANXI NORMAL UNIVERSITY

Outline

- 测试自动化的内涵
- 测试自动化实现的原理
- 测试自动化的实施
- 测试工具
 - 功能测试工具
 - 性能测试工具
 - 安全性测试工具
- 缺陷跟踪系统
- 管理工具

软件测试自动化

- 软件测试实行自动化进程，是测试工作的需要，即完成手工测试不能完成的任务，提高测试效率和测试结果的可靠性、准确性和客观性，提高测试覆盖率，保证测试工作的质量。

测试自动化的内涵

- 1 自动化测试（automated test）是相对手工测试（manual test）而存在的一个概念，由手工逐个地运行测试用例的操作过程被测试工具或系统自动执行的过程所代替，包括输入数据自动生成、结果的验证、自动发送测试报告等。
- 2 主要是通过所开发的软件测试工具、脚本（Script）等来实现，具有良好的可操作性、可重复性和高效率等特点。
- 3 测试自动化是软件测试中提高测试效率、覆盖率和可靠性等的重要手段，也就是说，测试自动化是软件测试不可分割的一部分。

测试自动化的内涵

- 1 手工测试的局限性
- 2 什么是测试自动化
- 3 软件测试自动化的优势

手工测试的局限性

- 通过手工测试无法做到覆盖所有代码路径，也难以测定测试的覆盖率。
- 通过手工测试很难捕捉到与时序、死锁、资源冲突、多线程等有关的错误。
- 在系统负载、性能测试时，需要模拟大量数据或大量并发用户等大负债的应用场景，没有测试工具的帮助时不可能实现的。

手工测试的局限性

- 在系统可靠性测试中，需要模拟系统运行几年、十几年，以验证系统能否稳定运行，也是手工测试无法模拟的。
- 在回归测试中，多数情况下时间很紧，希望一天能完成成千上万个测试用例的执行。
- 测试可以发现错误，但不能表明程序的正确性。

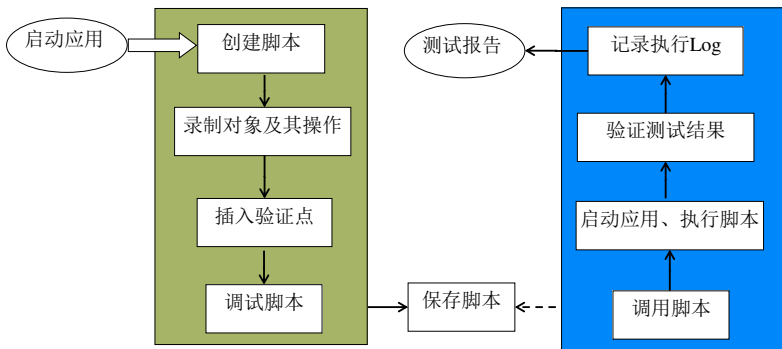
什么是测试自动化

- 1 自动化测试是把人为驱动测试行为转化为机器执行的一种过程，即模拟手工测试步骤，通过执行由程序语言编制的测试脚本，自动地完成软件的单元测试、功能测试、负载测试或性能测试等全部工作。
- 2 自动化测试集中体现着实际测试被自动执行的过程上，也就是由手工逐个地运行测试用例的操作过程被测试工具自动执行的过程所代替。

手工测试和自动化测试的区别

手工测试	自动化测试
发现缺陷率高	高效率（速度）
容易实施	高复用性
创造性、灵活性	覆盖率容易度量
覆盖率量化困难	准确、可靠
重复测试效率低	不知疲劳
不一致性、可靠性低	激励团队士气
依赖人力资源	机械、难以发现缺陷
	一次性投入大

自动化测试流程



软件测试自动化的优势

- 1 自动运行的速度快，执行效率高，是手工无法相比的。
- 2 永不疲劳
- 3 测试结果准确
- 4 可靠
- 5 可复用性
- 6 特别的能力

软件测试自动化的优势

- 1 缩短软件开发测试周期，软件自动化测试具有速度快，永远不知疲倦等特点；
- 2 更高的产品质量；
- 3 软件过程更规范；
- 4 测试效率高，充分利用硬件资源；
- 5 节省人力资源，降低测试成本；

软件测试自动化的优势

- 1 增强测试的稳定性和可靠性，通过测试工具运行测试软件，能保证百分之百进行；
- 2 提高软件测试的准确度和精确度，也就是提高测试的质量；
- 3 手工不能做的事情，软件测试工具可以完成；
- 4 高昂的团队士气，因为测试人员有更多机会学习编程、获取新技术，测试工作更有趣，有更多的挑战。

自动化测试的局限性

- 1 不能取代手工测试，不可能自动化所有的测试。如测试只是偶尔执行，或待测系统经常变动、不稳定，测试需要大量的人工参与时，就不适宜采用自动测试。
- 2 自动测试工具本身不具有想象力，只是按命令执行。而手工测试时测试执行者可以在测试中判断测试输出是否正确，以及改进测试，还可以处理意外事件。
- 3 自动测试对测试质量的依赖性较大，在确保测试质量的前提下，实施自动化测试才有意义。

自动化测试的局限性

- 1 自动测试在刚开始执行时，工作效率并不一定高于手动测试，只有当整个自动测试系统成熟，且测试工程师熟练掌握测试工具后，工作效率才会随着测试执行次数的增加而提高。
- 2 自动测试的成本可能高于手工测试。自动测试的成本大致有以下几个部分组成：自动测试开发成本、自动测试运行成本、自动测试维护成本和其他相关任务带来的成本。
- 3 软件的修改带来测试脚本部分或全部修改，就会增加测试维护的开销。

测试自动化实现的原理

- 1 软件测试自动化实现的基础是可以通过特定的程序（包括脚本、指令）模拟测试人员对计算机的操作过程及其行为
- 2 无论是对功能测试，还是对性能测试，自动化实现的方法都比较接近，只不过功能测试需要增加验证点，而性能测试不需要，但性能测试需要模拟成千上万的虚拟用户。

自动化测试的误区

- 1 期望自动化测试能够完全取代手动测试
- 2 期望自动化测试发现大量的新缺陷
- 3 期望自动化测试能够智能的完成绝大多数工作
- 4 期望自动化测试是一劳永逸的

自动化测试的目的

- 1 缩短测试周期，加快测试进度，从而加快产品发布进度
- 2 实现更大规模、更大频率的测试
- 3 减少手工测试的人力资源投入，降低测试成本
- 4 提高测试覆盖率
- 5 保证回归测试的可控性和一致性

自动化测试的目的

- 1 提高测试用例执行的可靠性
- 2 在不降低质量的情况下由低技能的人员完成
- 3 定义清晰的测试过程，降低和避免测试人员的个体对整体测试的影响
- 4 提高测试人员的工作效率，并使更高技能的人员有时间和资源，对产品进入更深层次的测试。
- 5 辅助测试人员完成手工无法完成的测试工作。

测试用例的质量指标

- 1 检测软件缺陷的有效性
- 2 测试用例的可仿效性
- 3 测试用例执行、分析和调试的经济性
- 4 测试用例的可修改性

适合做自动化测试情况

- 1 产品型项目
- 2 大型增量式开发、持续集成项目
- 3 能够自动编译、自动发布的系统
- 4 需要多次重复机械性动作的测试
- 5 简单而烦琐的基于命令行交互方式的测试
- 6 完成一些手工难以完成的测试目标
- 7 涉及大量第三方软件或设备的测试

不适合做自动化测试情况

- 1 按需定制型项目（通常为一次性的短期项目）
- 2 项目周期很短的项目
- 3 业务规则复杂的项目
- 4 依赖于人类的习惯、感官或智力的测试内容
- 5 不需要频繁测试的软件
- 6 软件不稳定

自动化测试和手工测试互为补充

- 1 在系统功能逻辑测试、验收测试、适用性测试、涉及交互性测试时，多采用手工测试方法；
- 2 单元测试、集成测试、系统负载或性能、可靠性测试等比较适合采用自动化测试；
- 3 对那种不稳定、开发周期短或一次性的软件等不适合自动化测试
- 4 工具本身缺乏想象力和创造性，自动测试只能发现 15% 的缺陷，而手工测试可以发现 85% 的缺陷；

测试自动化实现的原理

- 1 代码分析
- 2 对象识别
- 3 脚本技术
- 4 自动比较技术
- 5 测试自动化系统的构成

自动化测试的最佳实践

- 1 自动化测试的代码效率不是越高越好
- 2 自动化测试的执行过程和记录应便于分析
- 3 自动化测试的设计和脚本可读性越好越好
- 4 自动化测试的脚本可维护性是重中之重
- 5 完整性是自动化测试成功的前提
- 6 自动化测试脚本之间是松耦合或彼此独立的

测试自动化的实施

- 1 测试工具的分类
- 2 测试工具的选择
- 3 测试自动化普遍存在的问题
- 4 自动化测试的引入和应用

自动化测试的脚本技术

- 1 线性脚本
- 2 结构化脚本
- 3 共享脚本
- 4 数据驱动脚本
- 5 关键字驱动脚本

线性脚本

- 1 线性脚本是录制手工执行的测试事例得到的脚本。
- 2 如果用户只使用线性脚本技术，即录制每个测试事例的全部内容，则每个测试事例可以通过脚本完整地回放。
- 3 几乎任何重复的操作都可以使用线性脚本技术自动化。

线性脚本的优点

- 1 简单，只需坐在计算机前录制手工任务。
- 2 可以快速开始自动化。
- 3 对实际执行操作可以审计跟踪。
- 4 用户不必是编程人员（如果不修改）

线性脚本的缺点

- 1 无共享或重用脚本。
- 2 容易受软件变化的影响。
- 3 修改代价大。
- 4 如果回放脚本时发生了录制脚本时没有发生的事情，引起整个测试失败。

结构化脚本

- 1 结构化脚本类似于结构化程序设计，结构化脚本中含有控制脚本执行的指令。
- 2 控制脚本执行的指令：顺序，选择和迭代。

结构化脚本的优缺点

- 1 健壮性好，对一些容易导致测试失败的特殊情况进行处理。
- 2 可以执行许多其他类似的功能，如重复的指令可以使用循环结构。
- 3 可以作为模块被其他脚本调用。
- 4 脚本变得更加复杂，而且测试数据仍然‘捆绑’在脚本中。

共享脚本

- 1 脚本可以被多个测试用例使用。这意味着脚本语言允许一个脚本被另一个脚本调用，而这多少已成为所有测试执行自动化工具的标准。
- 2 这种技术思路是产生一个执行某种任务的脚本，而不同的测试要重复这个任务，当要执行这个任务时只需要在每个测试事例的适当地方调用这个脚本。

共享脚本的优点

- 1 以较少的开销实现类似的测试。
- 2 维护开销低于线性脚本。
- 3 删除明显的重复。
- 4 可以在共享脚本中增加更智能的功能。

共享脚本的缺点

- 1 需要跟踪更多的脚本，文档、名字、以及存储，很难找到适当的脚本。
- 2 对于每个测试仍需要一个特定的测试脚本。因此维护成本比较高。
- 3 共享脚本通常是针对被测软件的某个部分。

数据驱动脚本

- 1 数据驱动脚本技术将测试输入存储在独立的数据文件中，而不是存储在脚本中。
- 2 脚本中存放控制信息（如菜单导航）。执行测试时，从文件中而不是直接从脚本中读取测试输入。
- 3 这种方法的最大好处是同一个脚本可以运行不同的测试。

数据驱动脚本优点

- 1 可以很快增加类似的测试。
- 2 测试者增加新测试不必具有工具脚本语言的技术或编程知识。
- 3 对第二个测试及后续测试无额外的脚本维护开销。

数据驱动脚本缺点

- 1 初始建立的开销较大；
- 2 需要专业编程支持；
- 3 必须易于管理。

关键字驱动脚本

- 1 较复杂的数据驱动技术的逻辑扩展。
- 2 分为三层结构：一是控制脚本；二关键字动作描述；三是数据或测试用例。

测试工具

- 1 功能测试工具
- 2 性能测试工具
- 3 安全性测试工具

功能测试工具

- 1 关键特性
- 2 Selenium 及其应用
- 3 Web 功能测试工具
- 4 AutoIT 及其客户端测试工具
- 5 嵌入式测试工具

性能测试工具

- 1 JMeter 及其应用
- 2 性能测试工具的主要特征
- 3 常见的性能测试工具

安全性测试工具

- 1 安全测试工具没有绝对的标准
- 2 在选择安全性测试工具时，需要建立一套评估标准。根据这个标准，测试人员能够得到合适的且安全的工具，不会对软件开发和维护产生不利的影响。

缺陷跟踪系统

- 1 基于缺陷数据库
- 2 基于数据库系统
- 3 基于系统可以随时得到最新的缺陷状态
- 4 基于系统可以将缺陷和测试用例、需求等关联起来，可以完成更深度的分析，有利于产品的质量改进等。

管理工具

- 1 测试需求管理
- 2 编制测试计划
- 3 安排和执行测试
- 4 缺陷跟踪
- 5 人工与自动测试结合
- 6 图形化和报表输出
- 7 用户权限管理
- 8 和其他工具的集成

小结

- 测试自动化的内涵
- 测试自动化实现的原理
- 测试自动化的实施
- 测试工具
 - 功能测试工具
 - 性能测试工具
 - 安全性测试工具
- 缺陷跟踪系统
- 管理工具

小结

- 1 清楚何时使用工具和使用哪一种工具是软件测试员的重要技巧。
- 2 创建、使用测试工具和测试自动化是有意义的工作。

小结

- 1 测试工具的使用是自动化测试的主要特征，也是自动化测试的主要手段。
- 2 自动化测试，有时不需要测试工具，而是使用一些命令、Shell 脚本就可以完成测试任务；其次，自动化测试不能仅仅局限于工具本身，必须和测试目标和测试策略结合起来，包括了自动化测试的思想、流程和方法，在流程上支撑自动化测试的实现，在方法上保证选用正确的测试工具。

致谢

谢谢，欢迎提问！