

# 软件质量保证与测试

Software Quality Assurance and Testing

## 第 10 章 软件测试自动化



金陵科技学院

# 主要内容

- ◆ 软件测试自动化概述
- ◆ 自动化测试的原理
- ◆ 自动化测试的优势与局限
- ◆ 自动化测试的适用范围与注意事项
- ◆ 自动化测试的过程与方法
- ◆ 自动化测试工具介绍

# 自动化测试概述

测试自动化的必要性：

- ❑ 软件测试的工作量很大；
- ❑ 测试中的许多操作是重复性的、非智力性的和非创造性的，并要求准确细致；
- ❑ 有些测试工作人工难以完成

# 自动化测试概述

## 软件自动化测试的概念

它是相对手工测试而存在的，  
是通过开发的软件分析和测试工具、  
测试脚本等来实现软件分析和测试过程的自动化，具有良好的可操作性、可重复性和高效率等特点。



用户名

密 码

验证码  

自动化测试

# 自动化测试概述

## 自动化测试的优点

- 提高测试执行速度和测试效率，节省时间。
- 代替手工操作，节约人力资源，避免人的不良情绪
- 提高了测试的准确度和精确度。
- 能更好地利用资源。
- 能实现难以用人工完成的测试。

# 自动化测试概述

## 自动化测试的适用情况

- ✓ 回归测试；
- ✓ 大量不同数据输入的功能测试；
- ✓ 用手工测试完成难度较大的测试，如性能测试、负载测试、强度测试等。

# 自动化测试的认识误区

- 1、自动化测试可以完全取代人工测试
  - 测试用例的设计：测试人员的经验和对错误的猜测能力是工具不可替代的。
  - 界面和用户体验测试：人类的审美观和心理体验是工具不可模拟的。
  - 正确性的检查：人们对是非的判断和逻辑推理能力是工具不具备的。
- 2、测试用例可完全由测试工具自动生成；
- 3、测试工具可在任何场合使用；
- 4、自动化后测试效率立刻提高；

# 自动化测试的局限

- ❖ 下述情况下，自动化测试不太适用
  - ❧ 测试很少运行（不划算）
  - ❧ 软件不稳定（难以自动执行）
  - ❧ 需要通过人的感官来评判的测试（无法给出结论）
  - ❧ 涉及物理交互的测试（需要人的参与）



# 自动化测试的局限

- ❖ 自动化测试并不比手工测试发现的缺陷更多
- ❖ 应对自动测试程序进行正确性检查和验证
- ❖ 自动化测试对测试设计的依赖性极大
  - ❧ 测试工具本身只是起到辅助作用
  - ❧ 确保测试设计的质量更为关键

# 自动化测试的局限

- ❖ 自动化测试只是提高效率，而不能提高有效性
- ❖ 自动化测试可能会制约软件开发的改进
  - ❧ 比手工测试更加“脆弱”
  - ❧ 自动测试初期比手工测试开销大
  - ❧ 需要进行维护

# 自动化测试工具

测试工具可以从两个不同的方面去分类。

✓根据测试方法不同，自动化测试工具可以分为：

白盒测试工具、黑盒测试工具

✓根据测试的对象和目的，自动化测试工具可以分为：

单元测试工具、功能测试工具、负载测试工具、性能测试工具、Web测试工具、数据库测试工具、回归测试工具、嵌入式测试工具、页面链接测试工具、测试设计与开发工具、测试执行和评估工具、测试管理工具等。

# 主流测试工具

## 1、测试管理工具

- ▣ 用于对测试进行管理，负责对测试计划、测试用例、测试的实施进行管理。
- ▣ 还能对产品缺陷进行跟踪管理、产品特性管理等。
- ▣ 代表：IBM Rational公司的 TeamManager、  
HP Mercury Interactive公司的  
TestDirector(QualityCenter)；

# 主流测试工具

## 2、功能测试工具

用于检测程序能否达到预期的功能要求并正常运行。(QTP: QuickTest Professional)

## 3、性能测试工具

用于确定软件和系统的性能。(LoadRunner)

## 4、白盒测试工具(XUnit系列)

# 自动化测试的原理——黑盒测试

## （一）脚本技术

脚本是一组测试工具执行的指令集合，也是计算机程序的一种形式。脚本可以通过录制测试的操作产生，然后再做修改，这样可以减少脚本编程的工作量。当然，也可以直接用脚本语言编写脚本。例如以下为一段脚本：

```
startApp("ClassicsJavaA");  
.....  
placeOrder().click();  
rememberPassword().performTest(Password_textVP());  
ok().click();
```

# 自动化测试的原理——黑盒测试

## （一）脚本技术

脚本技术可以分为以下几类：

- ✓线性脚本——是录制手工执行的测试用例得到的脚本。
- ✓结构化脚本——类似于结构化程序设计，具有各种逻辑结构（顺序、分支、循环），而且具有函数调用功能。
- ✓数据驱动脚本、关键字驱动脚本、共享脚本。

# 自动化测试的原理——黑盒测试

## （二）录制和回放

❖ 录制是将用户每一步操作都记录下来。

记录操作位置（程序用户界面的像素坐标）或操作对象（窗口、按钮、滚动条等），以及相对应的操作（输入、事件触发、状态变化或是属性变化）。

所有的记录转换为一种脚本语言所描述的过程（指令集）。



# 自动化测试的原理——黑盒测试

## （二）录制和回放

回放时，将脚本语言所描述的过程转换为屏幕上的操作，然后将被测系统的输出记录下来同预先给定的标准结果比较。

这样可以大大减轻黑盒测试的工作量，在迭代开发的过程中，能够很好地进行回归测试。

# 自动化测试的原理——白盒测试

## ❖ 代码分析

代码分析类似于高级编译系统，一般针对不同的高级语言去构造分析工具，在工具中定义类、对象、函数、变量等定义规则、语法规则；在分析时对代码进行语法扫描，生成系统的调用关系图，找出不符合编码规范的地方；并可以根据某种质量模型评价代码质量等。

# 适合自动化的测试活动

- ❖ 自动生成部分测试用例
- ❖ 重复执行测试过程
- ❖ 测试结果与标准输出的对比
- ❖ 不匹配的测试结果的分析、记录、分类
- ❖ 测试状况的统计，报表的产生

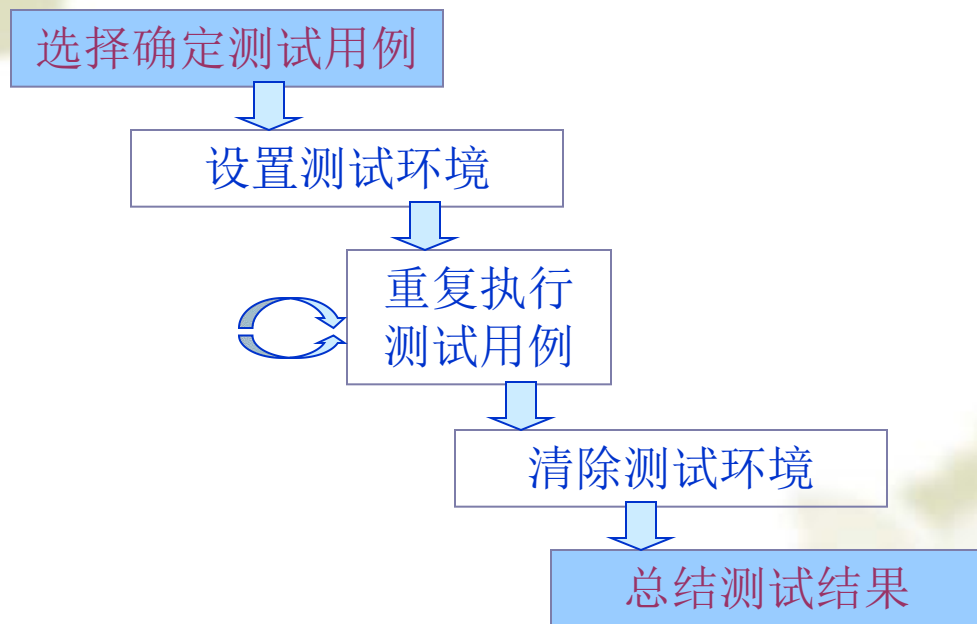
# 使用自动测试应注意的问题

- ❖ 容易产生不现实的期望
  - ✎ 测试工具不能解决所有的问题；
  - ✎ 对测试工具寄予过高的期望，最终无法实现；
- ❖ 缺乏测试实践经验
  - ✎ 当缺乏实践经验、测试组织较差、文档较少或不一致时，不要盲目建立大型自动测试

# 使用自动测试应注意的问题

- ❖ 自动化测试的可维护性
- ❖ 技术问题
  - ❧ 测试工具本身的技术问题
  - ❧ 被测软件本身的可测试性
- ❖ 组织问题
  - ❧ 应该进行工具选型、培训、实践并在组织内确立自动测试标准
  - ❧ 转变工作效率的观念

# 自动化测试的过程



# 自动化黑盒测试的过程

- ❖ 设置测试环境
  - ❧ 创建测试环境
  - ❧ 装载测试数据
- ❖ 执行测试
  - ❧ 脚本录制和优化
  - ❧ 自动化测试执行
  - ❧ 记录结果
- ❖ 清除测试环境
- ❖ 测试总结

# 自动化黑盒测试要点难点

## ❖ 过程要点

- ❧ 录制

- ❧ 验证点 (结果验证、同步等)

- ❧ 数据驱动

- ❧ 关闭被测软件，停止录制

- ❧ 脚本优化

- ❧ 批处理文件

- ❧ 回放和日志



# 自动化黑盒测试要点难点

## ❖ 难点

☞ 验证点 (结果验证、同步等)

```
startApp("ClassicsJavaA");
```

```
//结果验证
```

```
placeOrder().click();
```

```
rememberPassword().performTest>Password_textVP());
```

```
ok().click();
```

# 自动化黑盒测试要点难点

## ❖ 难点

☞ 验证点 (结果验证、同步等)

//同步

```
tree2().click(atPath("Composers->Bach->Location(PLUS_MINUS)"));
```

```
tree2().click(atPath("Composers->Bach->Violin Concertos"));
```

```
placeOrder().click();
```

```
placeOrder2().performTest(okButtonPropertiesVP());
```

```
ok().click();
```

# 自动化测试工具介绍

## ❖ 流行软件测试工具

☞ Rational 系列: Logiscope 、 RFT、 RPT

☞ WinRunner

☞ LoadRunner

☞ Junit