



提纲

- ◆ 软件测试管理概述
- ◆ 构建测试环境（测试准备）

1. 软件测试管理概述

- ◆ 有了测试的原则、技术、阶段等测试基本内容的掌握，是否就可以进行测试工作并取得成功呢？
- ◆ 这个问题与“学习了编程语言和软件编码技术，是否就可以开发出好的软件”非常类似

1.1 测试管理的基本内容

- ◆ 管理中的戴明环：一件工作的完成可以分为4个部分：**计划→执行→检查→处理**
- ◆ 软件测试管理中的戴明环：

```
graph TD; 0[0: 测试准备] --> 1[1. 测试计划]; 1 --> 2[2. 测试执行]; 2 --> 3[3. 测试检查]; 3 --> 4[4. 测试处理]; 4 --> 1;
```

1.2 软件测试管理概念

- ◆ **软件测试管理 (Software Testing management)** ——是指软件测试的组织、标准流程、标准文档以及培训等一系列保证软件测试有效性的管理手段
- ◆ 测试管理之于测试，相当于软件工程之于软件开发

1.3 测试管理工具

- ◆ 完成测试全过程的管理：测试**需求**管理、测试**计划**管理、测试**用例**管理、测试**套件**管理、测试**执行**管理、测试**版本**管理、**缺陷**管理、**发布**管理和**分析报表**。
- ◆ 华为软开云
- ◆ BugZilla + Testopia
<https://www.bugzilla.org/> 开源
<http://www.mozilla.org/projects/testopia/> 在BugZilla上扩展的测试用例管理，开源

2. 构建测试环境

构建测试环境是在软件测试前的准备工作

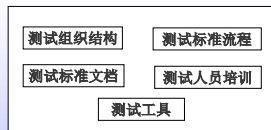
- ◆ 测试环境概述
- ◆ 测试环境组成
 - 测试组织结构
 - 标准测试流程
 - 测试文档
 - 测试人员的培训
 - 测试工具

2.1 测试环境概述

- ◆ 测试环境是测试管理和测试过程的基础
- ◆ 一方面，测试环境中的组织、目标、任务和策略作为测试管理的前提和依据
- ◆ 另一方面，测试环境又是启动测试过程的基础，在测试环境构建好之后才可以启动测试过程

2.2 测试环境的组成

- ◆ 测试环境包括测试所需的要素



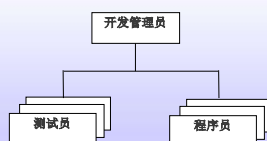
测试环境的组成

2.3 测试组织结构

- ◆ 构建测试环境的首要任务是建立测试组织结构
- ◆ 按照测试的独立性原则，构建的测试组织也应该具有独立性，即独立于开发人员
- ◆ 测试组织的几种结构形式
 - 融合式测试组织结构
 - 依附式测试组织结构
 - 独立式测试组织结构

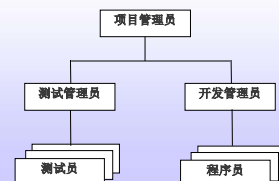
2.3.1 融合式测试组织结构

- ◆ 融合式测试组织结构将测试团队置身于开发团队之内
- ◆ 适合小型软件系统



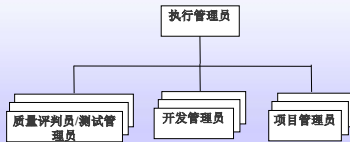
2.3.2 依附式测试组织结构

- ◆ 测试团队与开发团队同级，同属于一个项目组织下
- ◆ 这种结构适合于中型的软件项目



2.3.3 独立式测试组织结构

- ◆ 高级别的独立质量团队，监督和控制**所有项目**的质量水平
- ◆ 适合大中型软件或可靠性、安全性要求很高的软件



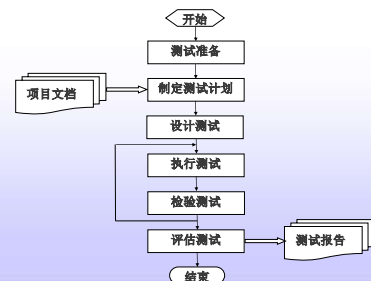
比较：测试组织



2.4 标准测试流程

- ◆ **软件测试流程**是指从测试开始到测试结束为止的一系列准备、执行及分析的过程
- ◆ 软件测试流程的优劣决定了测试结果的好坏。因此，测试工作也是需要标准流程进行指导的

2.4.1 标准的软件测试流程



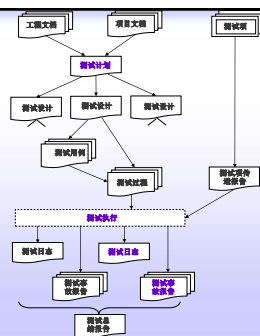
2.4.2 软件测试流程的保证

- ◆ 标准的测试流程需要有文档进行保证，每个阶段的文档应包括的内容：
 1. **输入**：完成每个任务所需要的入口标准或交付产品
 2. **执行规程**：将输入转换成输出的工作任务或过程
 3. **检验规程**：确定输出是否符合标准的规则
 4. **输出**：这个过程中产生的出口标准或交付产品。

2.5 测试文档

- ◆ 测试的标准过程是由众多的测试标准文档进行保证的
- ◆ IEEE Std 820-1998规定了测试工作中所需要的各种标准文档

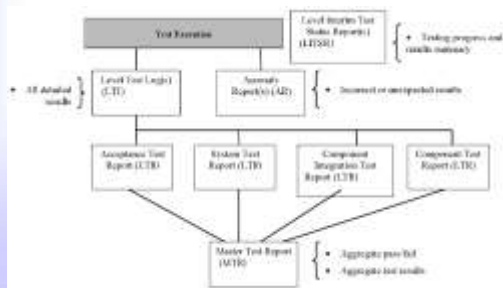
测试的标准文档



IEEE Std 829-2008中的测试文档



IEEE Std 829-2008中的测试文档



2.6 测试人员的培训

- 为了有效地测试，需要测试人员能够完全有能力掌握测试中需要使用到的测试技术
- 测试人员的能力分级：测试人员对于测试知识的掌握程度分为3级：
 - 无能力：不具有该技能或不能够将该技术用于进行测试
 - 有能力：已经学习了测试技能，但没有充分的实践，从而不能认为自己已经完全掌握了这种技能
 - 完全有能力：不仅理解测试技能，还知道如何使用这项技能，对于有效执行技能非常有信心

2.7 测试工具

- 工具的使用可以帮助人们提高劳动效率，使人具有更多思考问题的时间
- 同样地，测试工具的使用帮助测试人员提高测试的效率和减轻测试人员的劳动，因此需要构建相应的测试工具集

常见的测试工具分类

- 捕获/重放
- 检查表
- 控制流分析
- 数据流分析
- 探测
- 并行模拟
- 测试数据生成器
- 测试脚本
- 跟踪

2.8 其他测试管理

- ◆ 配置管理
- ◆ 风险管理
- ◆ 软件测试的成本管理

2.8.1 配置管理

- ◆ 配置管理的目的是在项目和产品的生命周期内建立和维护软件或系统产品（组件、数据和文档）的完整性
- ◆ 配置管理的作用
 1. 识别出所有测试件（testware），控制其版本
 2. 在测试文档中，可以明确所有待测的文档和测试项

2.8.2 风险管理

- ◆ 风险（Risk）——是指发生事件、危险、威胁以及由此产生不可预测后果的可能性，即潜在的问题
 - 项目风险
 - 产品风险
- ◆ 风险级别由风险发生的可能性和发生后产生的影响两个方面来决定

2.8.2.1 项目风险（Project Risk）

- ◆ 项目风险（Project Risk）是指项目是否能够按要求交付软件产品的风险，包括：
 1. 供应商的问题
 2. 公司组织因素
 3. 技术方面的问题

2.8.2.2 产品风险（Product Risk）

- ◆ 产品风险（Product Risk）在软件或系统中潜在失效的区域。产品风险对产品质量而言是一个风险，包括：
 1. 易错（error-prone）的软件交付使用
 2. 软件或硬件可能会对个人或公司造成伤害的可能性
 3. 劣质的软件特征，比如功能性、安全性、可靠性、可用性和性能等
 4. 软件没有实现既定的功能等

2.8.3 测试成本管理

- ◆ 软件成本构成
 - 测试初始成本
 - 测试阶段修复故障的成本
 - 测试阶段测试资源所花费的成本
 - 操作运行阶段修复故障的成本
 - 由于软件失效所导致的成本
 - 由软件延期发布或交付引发的成本

测试资源、测试成本、软件发布关系

