



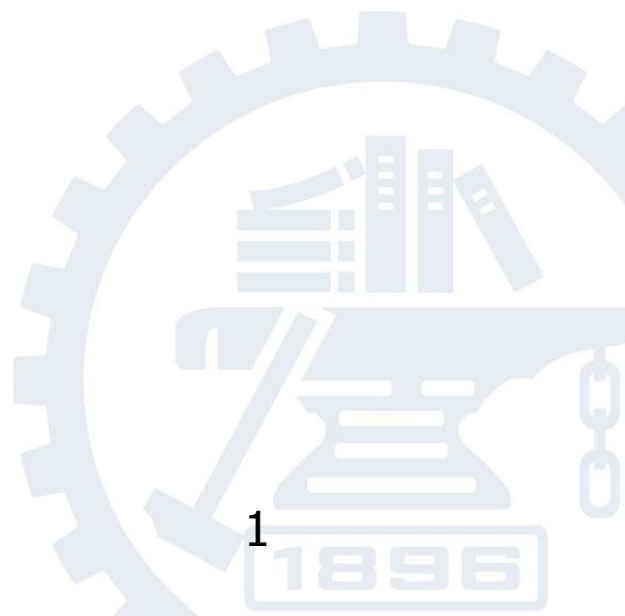
北京交通大学
BEIJING JIAOTONG UNIVERSITY



高级软件测试技术

2020年9月

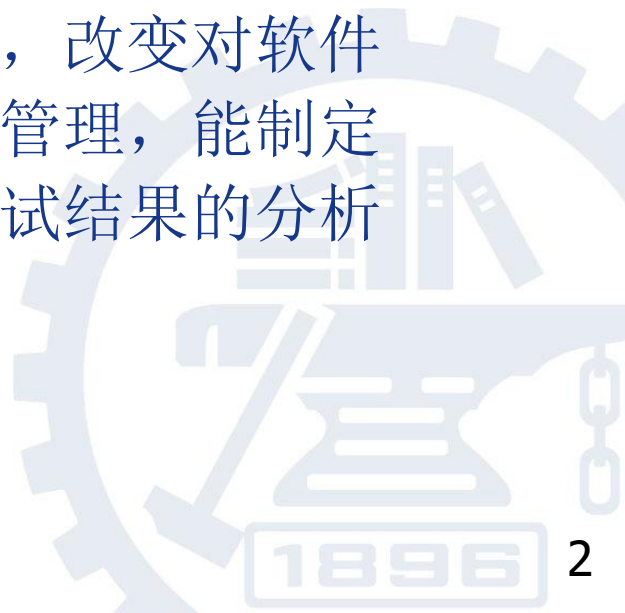
hhli@bjtu.edu.cn





课程内容

- 软件测试是保证软件可靠性和软件质量的重要手段，对软件开发和维护有很强的针对性和实用性。
- 课程主要讲述软件测试的基本理论、策略、标准、方法和常用工具。
- 通过本课程的学习，希望同学们掌握软件测试的基本概念，方法、技术和常用工具的使用，改变对软件测试的错误认识，对测试能进行有效的管理，能制定软件测试计划，设计测试用例，进行测试结果的分析和报告的书写。





课程内容

- ① 第一章 软件测试概述
- ② 第二章 软件测试策略与过程
- ③ 第三章 动态测试技术
- ④ 第四章 静态测试技术
- ⑤ 第五章 软件测试自动化
- ⑥ 第六章 软件测试管理



参考书

- ① 《高级软件测试技术》 杜庆峰 著 清华大学出版社
- ② 《软件测试方法和技术》 朱少民 清华大学出版社
- ③ 《软件缺陷模式与测试》 宫云战 科学出版社
- ④ 《软件测试》 (美) RON PATTON 张小松跃等译 机械工业出版社



第一章 软件测试概述

- ④ 1.1 软件测试背景
- ④ 1.2 软件测试基础理论
- ④ 1.3 软件开发与软件测试的关系
- ④ 1.4 软件测试过程
- ④ 1.5 软件质量保证
- ④ 1.6 软件测试职业





1.1 软件测试背景

1.1.1 软件可靠性问题

1.1.2 软件缺陷与故障

1.1.3 软件缺陷产生的原因

1.1.1 软件可靠性问题

- ④ 因软件设计故障与因计算机硬件设计故障而引发的系统失效的比例大约是：10:1
- ④ 运行软件的驻留故障密度（每千行代码的故障数目）：
 - 要求很高的关键财务或财产软件为：每千行代码1~10个故障
 - 关键的生命软件为：每千行代码0.01~1个故障
- ④ IEEE将软件可靠性定义为：系统在特定环境下，在给定的时间内无故障运行的概率。
- ④ 软件可靠性是对软件在设计、开发以及所预定的环境下具有能力的置信度的一个度量，是衡量软件质量的主要参数之一。而软件测试则是保证软件质量、提高软件可靠性的最重要手段。

1.1.1 软件可靠性问题

- ④ 软件可靠性 (software reliability)是软件产品在规定的条件下和规定的时间区间完成规定功能的能力。
 - 规定的条件是指直接与软件运行相关的使用该软件的计算机系统的状态和软件的输入条件，或统称为软件运行时的外部输入条件；
 - 规定的时间区间是指软件的实际运行时间区间；
 - 规定功能是指为提供给定的服务，软件产品所必须具备的功能。
- ④ 软件可靠性不但与软件存在的缺陷和（或）差错有关，而且与系统输入和系统使用有关。软件可靠性的概率度量称软件可靠度。

1.1.2 软件缺陷与故障

1、软件缺陷和软件故障案例

- 案例1 跨世纪“千年虫”问题
- 案例2 Windows 2000 中文输入法漏洞
- 案例3 金山词霸bug

实例中的软件问题在软件工程或软件测试中都被称为**软件缺陷或软件故障**。

1.1.2 软件缺陷与故障

1、软件缺陷和软件故障案例

- 案例4 铁路“7.23”事故
- 案例5 铁路“12306”早期缺陷
- 案例6 奥运会售票系统

1.1.2 软件缺陷与故障

2、软件缺陷的定义 (符合下列规则的叫软件缺陷)

- (1) 软件未达到产品说明书中已经标明的功能;
- (2) 软件出现了产品说明书中指明不会出现的错误;
- (3) 软件未达到产品说明书中虽未指出但应当达到的目标;
- (4) 软件功能**超出了**产品说明书中指明的范围;
- (5) 软件测试人员认为软件难以理解、不易使用, 或者最终用户认为该软件使用效果不良。

软件缺陷是指在软件系统中会导致系统不能实现其功能的缺陷 (包括Defect或Bug)。

1.1.2 软件缺陷与故障

3、软件缺陷的特征

- “看不到”

——软件的特殊性决定了缺陷不易看到

- “看到但是抓不到”

——发现了缺陷，但不易找到问题发生的原因所在

1.1.3 软件缺陷产生的原因分布

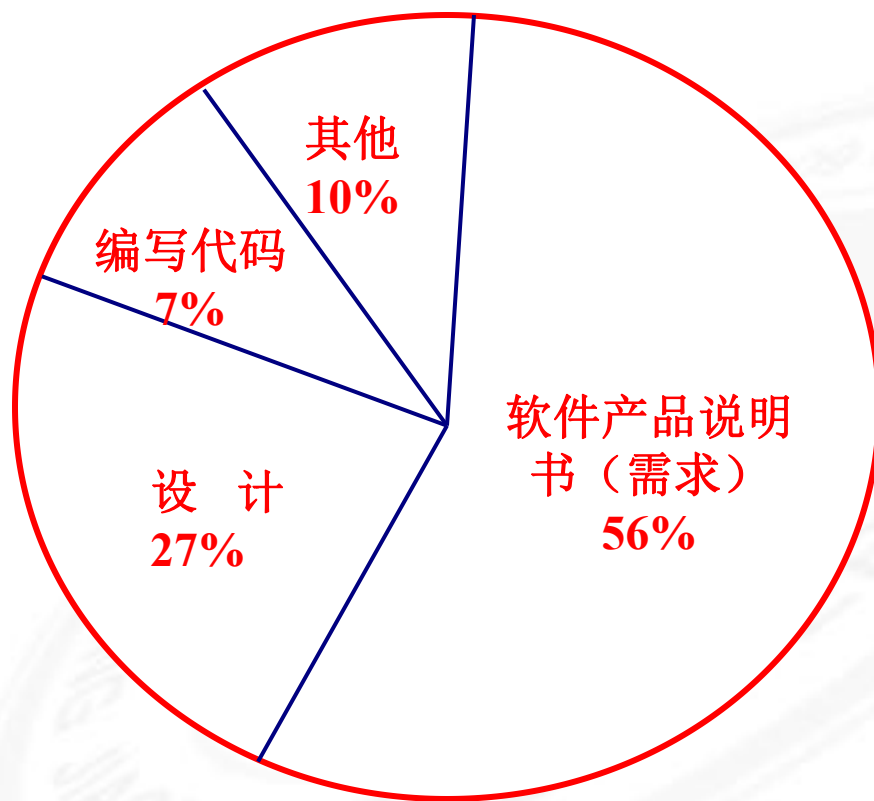


图1-1 软件缺陷产生的原因分布



1.2 软件测试基础理论

1.2.1 软件测试的定义

1.2.2 软件测试的基本理论

1.2.3 软件测试和缺陷修复的代价

1.2.4 软件测试技术概要

1.2.1 软件测试的定义

1、软件测试的定义

软件测试是贯穿整个软件开发生命周期、对软件产品（包括阶段性产品）进行验证和确认的活动过程，其目的是尽快尽早地发现在软件产品中所存在的各种问题——与用户需求、预先定义的不一致性。


通常对软件测试的定义有两种描述：



1.2.1 软件测试的定义

1、软件测试的定义

描述一： 软件测试是根据软件开发各阶段的规格说明和程序的内部结构而精心设计的一批测试用例，并利用这些测试用例运行程序以及发现错误的过程，即**执行测试步骤的过程**。

 **描述二：** 软件测试是一项活动，在这项活动中某个系统或组成的部分将在特定的条件下运行，结果将被观察和记录，并对系统或组成部分进行评价。测试活动有两种结果：找出缺陷和故障，或显示软件执行正确。测试是一个或多个测试用例的集合。

1.2.1 软件测试的定义

- **测试用例：**测试用例是为特定的目的而设计的一组测试输入、执行条件和预期的结果；测试用例是执行测试的**最小**实体。
- **测试步骤：**测试步骤详细规定了如何设置、执行、评估特定的测试用例。

1.2.1 软件测试的定义

- ④ 软件生命周期：一个软件生命周期包括制定计划、需求分析定义、软件设计、程序编码、软件测试、软件运行、软件维护、软件停用等8个阶段。
- ④ 软件测试的对象：
 - 软件测试不等于程序测试。
 - 软件测试贯串于软件定义和开发的整个过程。
 - 软件开发过程中所产生的需求规格说明、概要设计规格说明、详细设计规格说明以及源程序都是软件测试的对象。



测试的对象

- 软件测试目的：发现程序及前期开发的错误

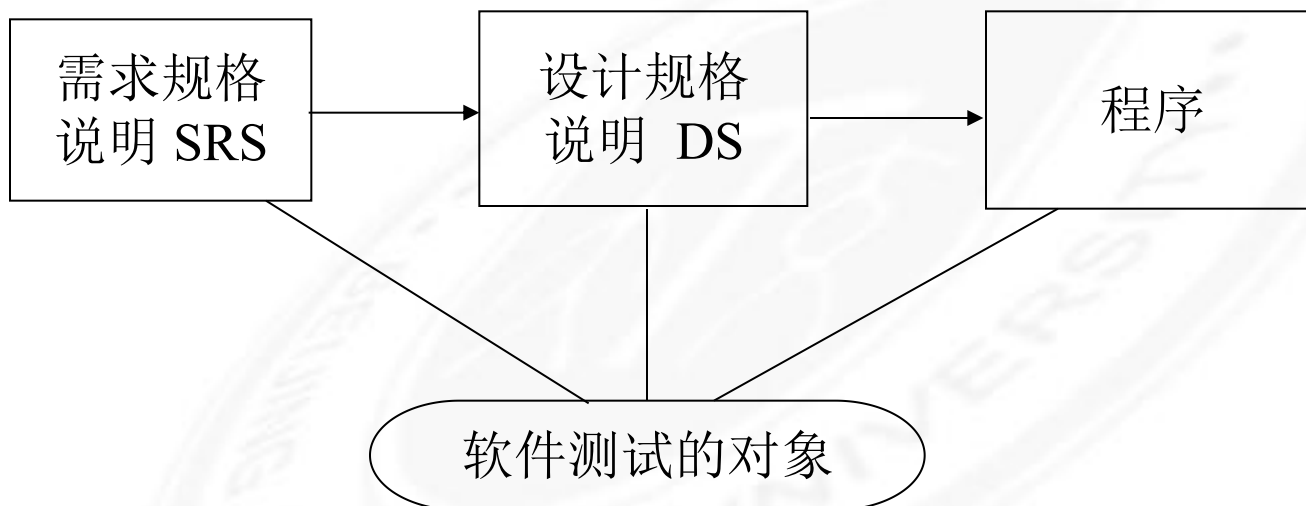


图1-2 测试对象



软件错误产生原因:

- 开发人员不太了解需求
- 软件系统越来越复杂，开发人员不太可能精通所有的技术
- 技术文档普遍比较糟糕，文档本身就有错误
- 软件需求、设计报告、程序经常发生变更，每次变更都可能产生新的错误
- 任何人在编程时都可能犯错误，导致程序中有错误
- 人们常处于进度的压力之下，急忙之下容易产生错误
- 人们过于自信，不真实的“没问题”将产生真正的问题

1.2.2 软件测试的基本理论

1、软件测试的目的

- (1) 测试是程序的执行过程，目的在于发现错误；不能证明程序的正确性，除非仅处理有限种情况。
- (2) 检查系统是否满足需求也是测试的期望目标。
- (3) 一个好的测试用例在于发现了还未曾发现的错误；一次成功的测试则是发现了错误的测试。

测试无法说明错误不存在，只能说明软件错误已出现。

1.2.2 软件测试的基本理论

2、软件测试的原则

- (1) 尽早地和及时地测试;
- (2) 测试用例应当由测试数据和与之对应的预期结果这两部分组成;
- (3) 在程序提交测试后, 应由专门的测试人员进行测试;
- (4) 测试用例应包括合理的输入条件和不合理的输入条件;
- (5) 严格执行测试计划, 排除测试的随意性;
- (6) 应对每一个测试结果做全面的检查;
- (7) **80-20原则**: **80%**的错误聚集在**20%**的模块中, 经常出错的模块改错后还会经常出错;
- (8) 保存测试计划、测试用例、出错统计和最终分析报告, 为维护工作提供充分的资料。

1.2.2 软件测试的基本理论

3、软件测试的分类

软件测试按照不同的划分方法，有不同的分类：

- 广义上说，可分为功能性测试和非功能性测试。
- 按照软件测试用例的设计方法而论，软件测试可以分为白盒测试法、黑盒测试法和灰盒测试。
- 按照软件测试阶段来分类，软件测试可分为单元测试、集成测试、系统测试、验证测试和确认测试。

1.2.2 软件测试的基本理论

3、软件测试的分类（续）

- 按照测试的方式（是否运行被测试软件），软件测试可分为：**静态测试和动态测试。**

1.2.2 软件测试的基本理论

4、测试信息流程

测试过程中需要三类输入：软件配置、测试配置和测试工具。

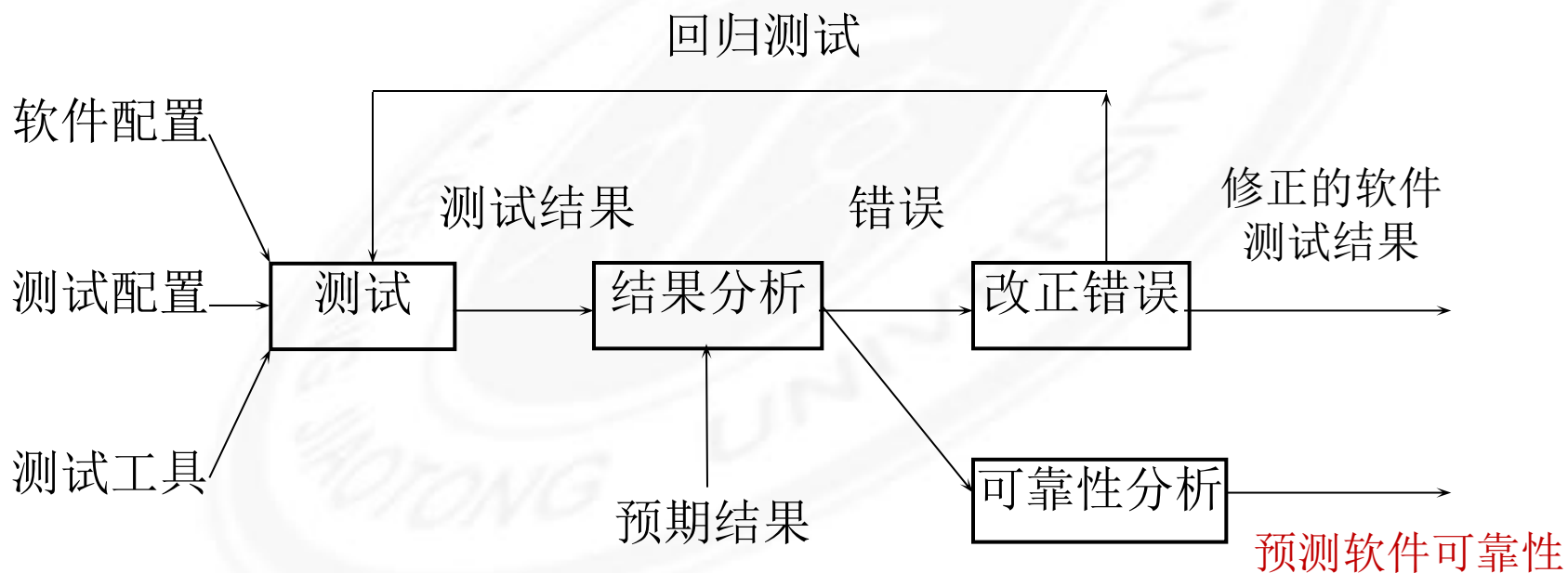


图1-3 测试信息流程

1.2.2 软件测试的基本理论

5、软件测试的周期性

软件测试的周期性是“测试->改错->再测试->再改错”这样一个循环过程，如下图1-4所示。

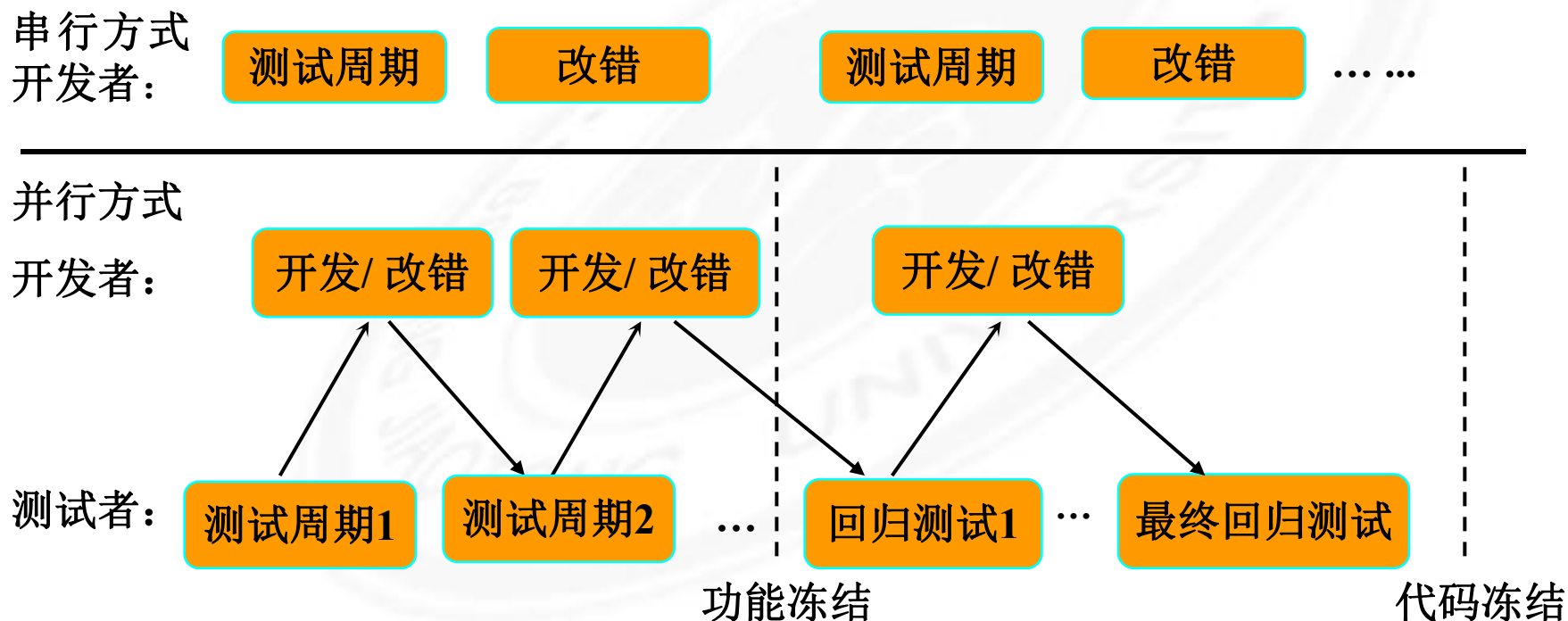


图1-4 软件测试的周期性

1.2.2 软件测试的基本理论

6、测试停止的依据（标准）

- **第一类标准：**测试超过了预定时间，则停止测试。
- **第二类标准：**执行了所有的测试用例，但并没有发现缺陷，则停止测试。
- **第三类标准：**使用特定的测试用例设计方案作为判断测试停止的基础。
- **第四类标准：**正面指出停止测试的具体要求，即停止测试的标准可定义为查出某一预订数目的故障。
- **第五类标准：**根据单位时间内查出故障的数量决定是否停止测试。

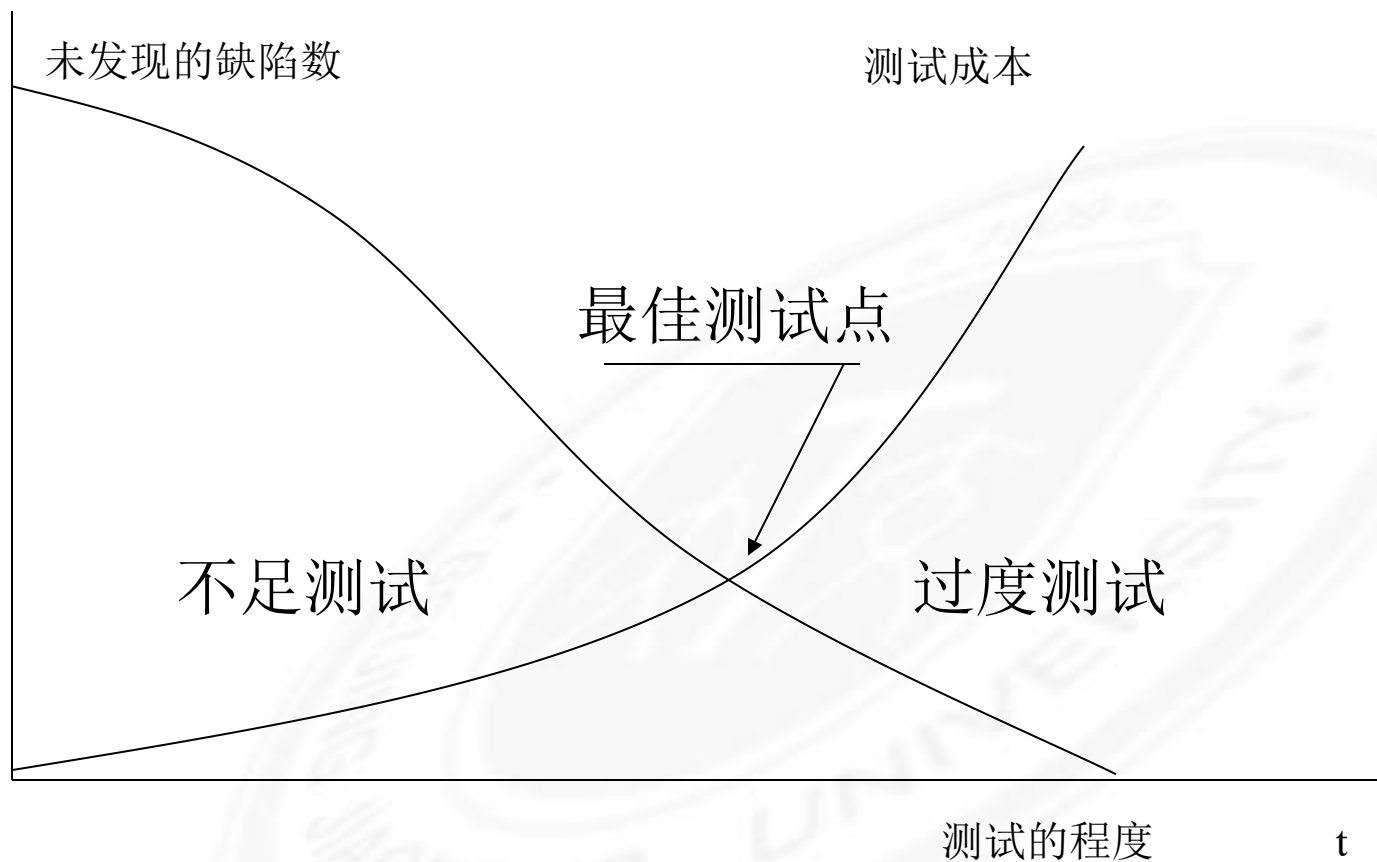


图1-5 测试成本曲线

1.2.3 软件测试和缺陷修复的代价

- 软件在从需求、设计、编码、测试一直到交付用户公开使用后的过程中，都有可能产生和发现缺陷。随着整个开发过程的时间推移，更正缺陷或修复问题的费用呈几何级数增长。

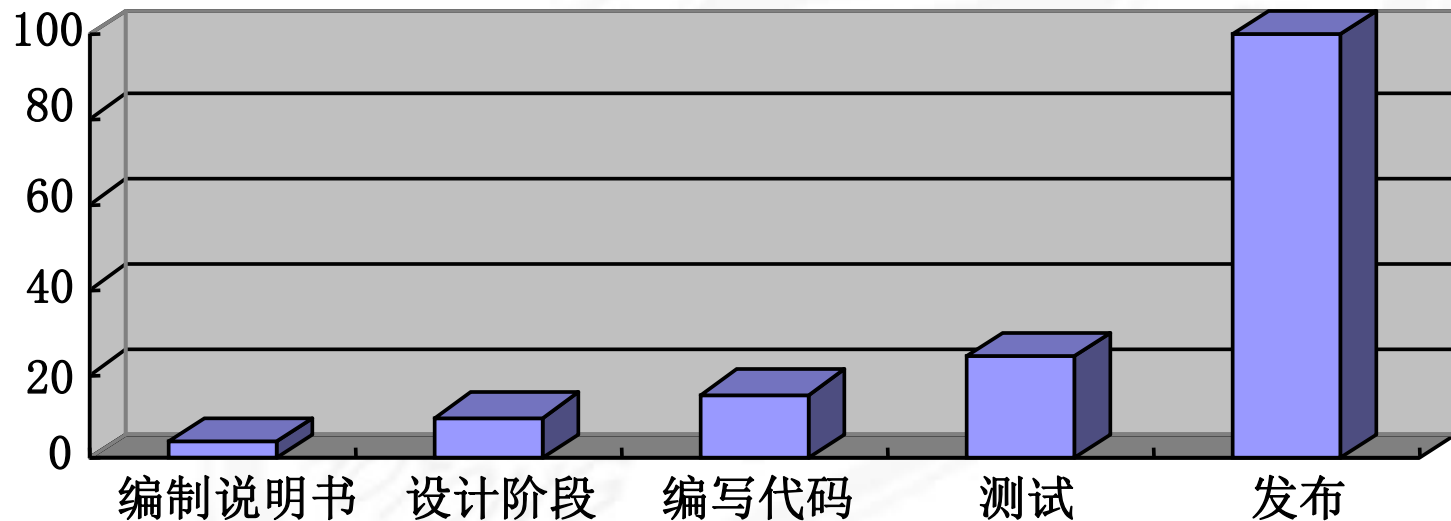


图1-6 软件缺陷在不同阶段发现时修复的费用示意图

IBM的研究结果表明，缺陷存在放大趋势。

- 下图表示了缺陷放大模型大致状况：



- 由此可见，问题发现越早，解决问题的代价就越小，这是软件开发过程中的黄金法则。

1.2.4 软件测试技术概要

- ④ 软件测试策略：指测试将按照什么样的思路和方式进行。通常，软件测试要经过单元测试、集成测试、确认测试、系统测试以及验收测试。
- ④ 主要软件测试技术：
 - (1) 白盒测试和黑盒测试
 - (2) 静态测试和动态测试
 - (3) 传统测试方法和面向对象测试的方法
 - (4) 特定环境及应用的测试

1.2.4 软件测试技术概要

● 软件测试技术的研究热点领域：

- (1) 软件验证技术
- (2) 静态测试分析技术
- (3) 测试数据的选择——主要对测试用例进行选择
通常从下面几个方面评价测试用例的质量：
 - 检测软件缺陷的有效性、测试用例的可重用性、测试用例的经济性、测试用例的可维护性
- (4) 集成化测试——研究如何实现软件测试的自动化过程以及相关的一系列内容。

1.3 软件开发与软件测试的关系

1、测试与开发各阶段的关系

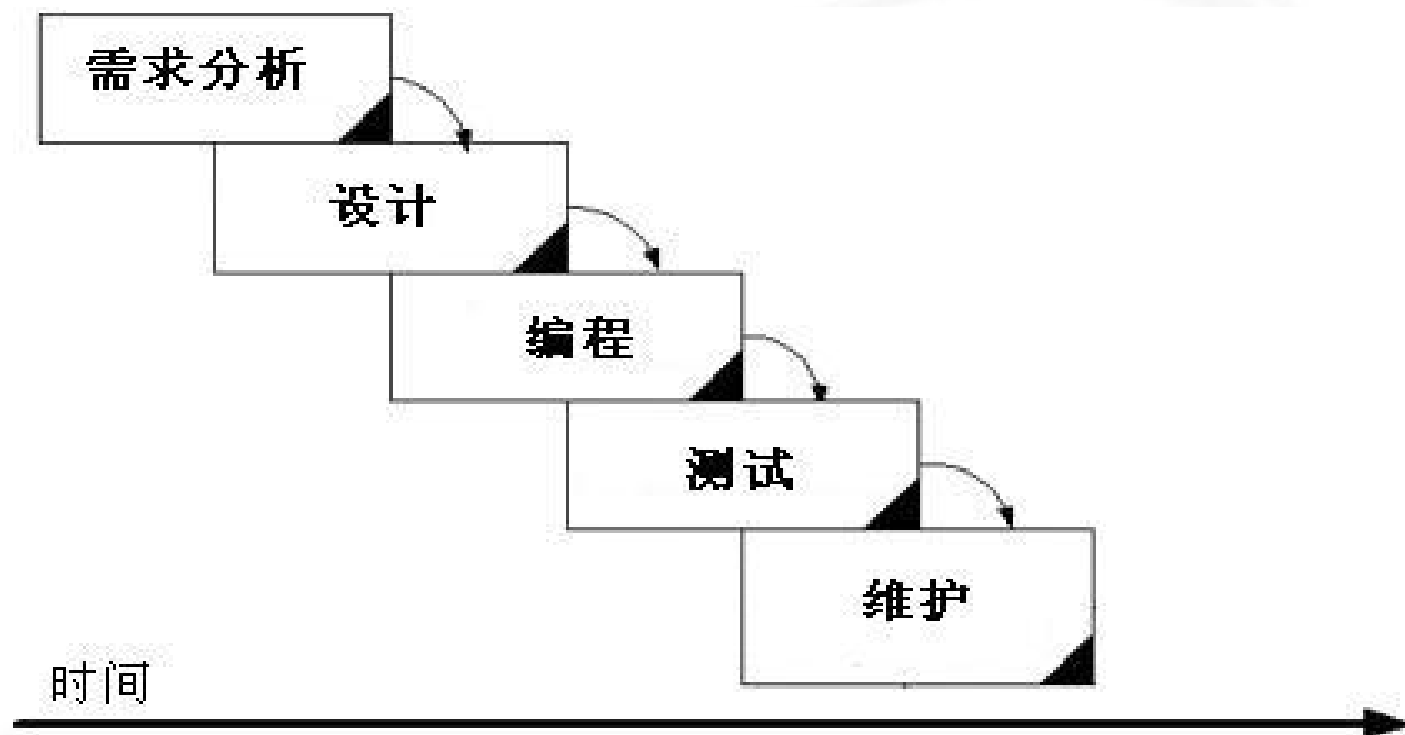


图1-7 软件开发基本过程

1.3 软件开发与软件测试的关系

1、测试与开发各阶段的关系

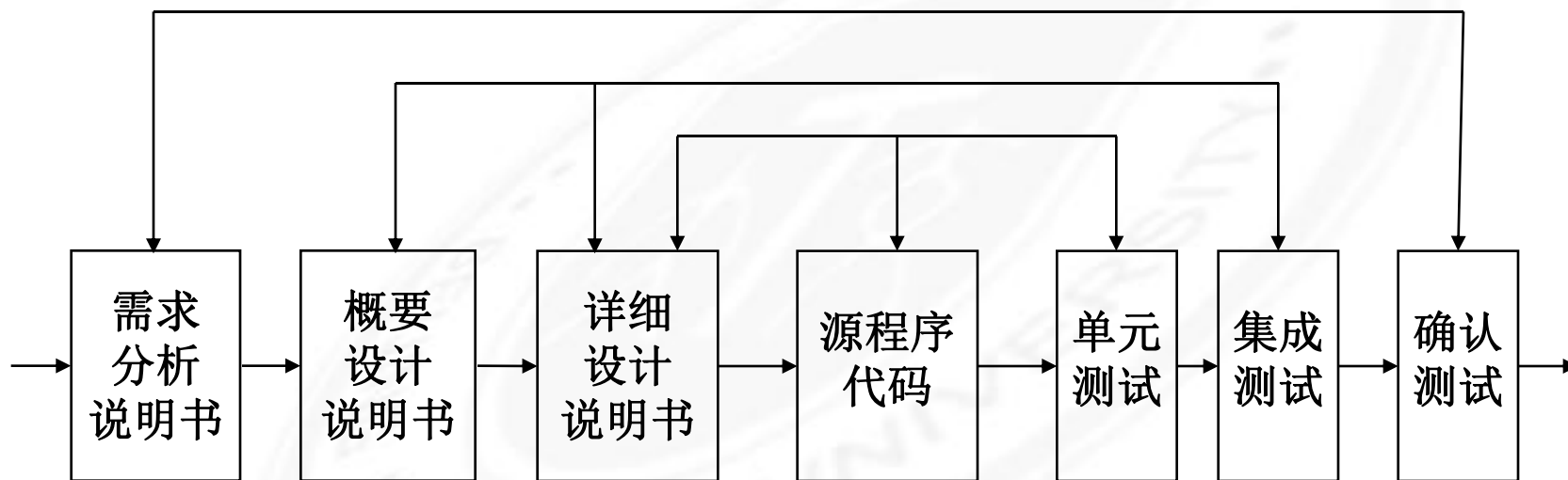


图1-8 软件测试与软件开发过程的关系

1.3 软件开发与软件测试的关系

测试在软件开发阶段的作用：

- **项目规划阶段**：负责从单元测试到系统测试的整个测试阶段的监控。
- **需求分析阶段**：确定测试需求分析、系统测试计划的制定，评审后成为管理项目。
- **详细设计和概要设计阶段**：确保集成测试计划和单元测试计划完成。
- **编码阶段**：由开发人员进行自己负责部分代码的测试。在项目较大时，由专人进行编码阶段的测试任务。
- **测试阶段**（单元、集成、系统测试）：依据测试代码进行测试，并提交相应的测试状态报告和测试结束报告。

1.3 软件开发与软件测试的关系

2、测试与开发的并行性

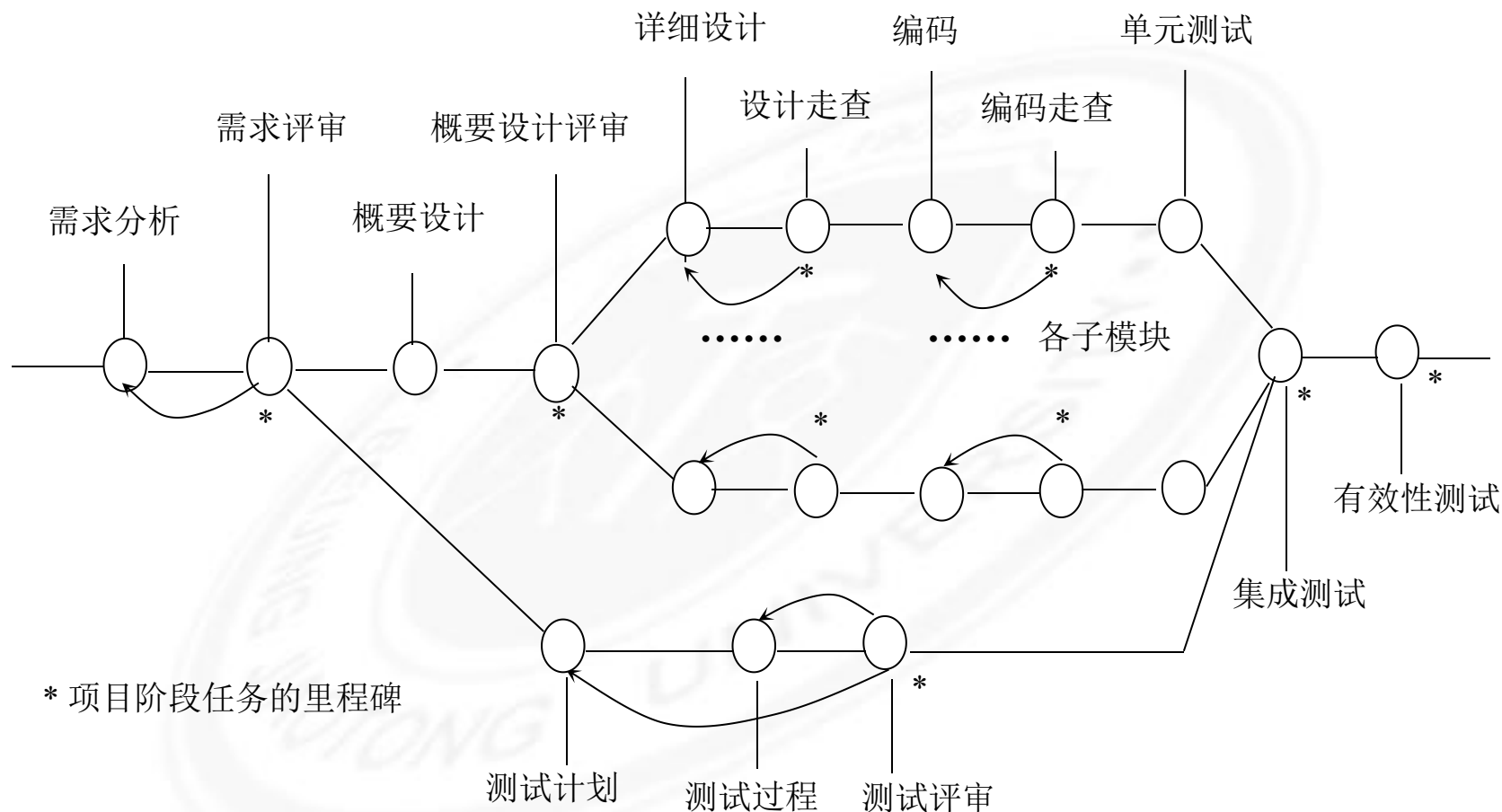


图1-9 软件测试与软件开发的并行性

软件测试过程模型

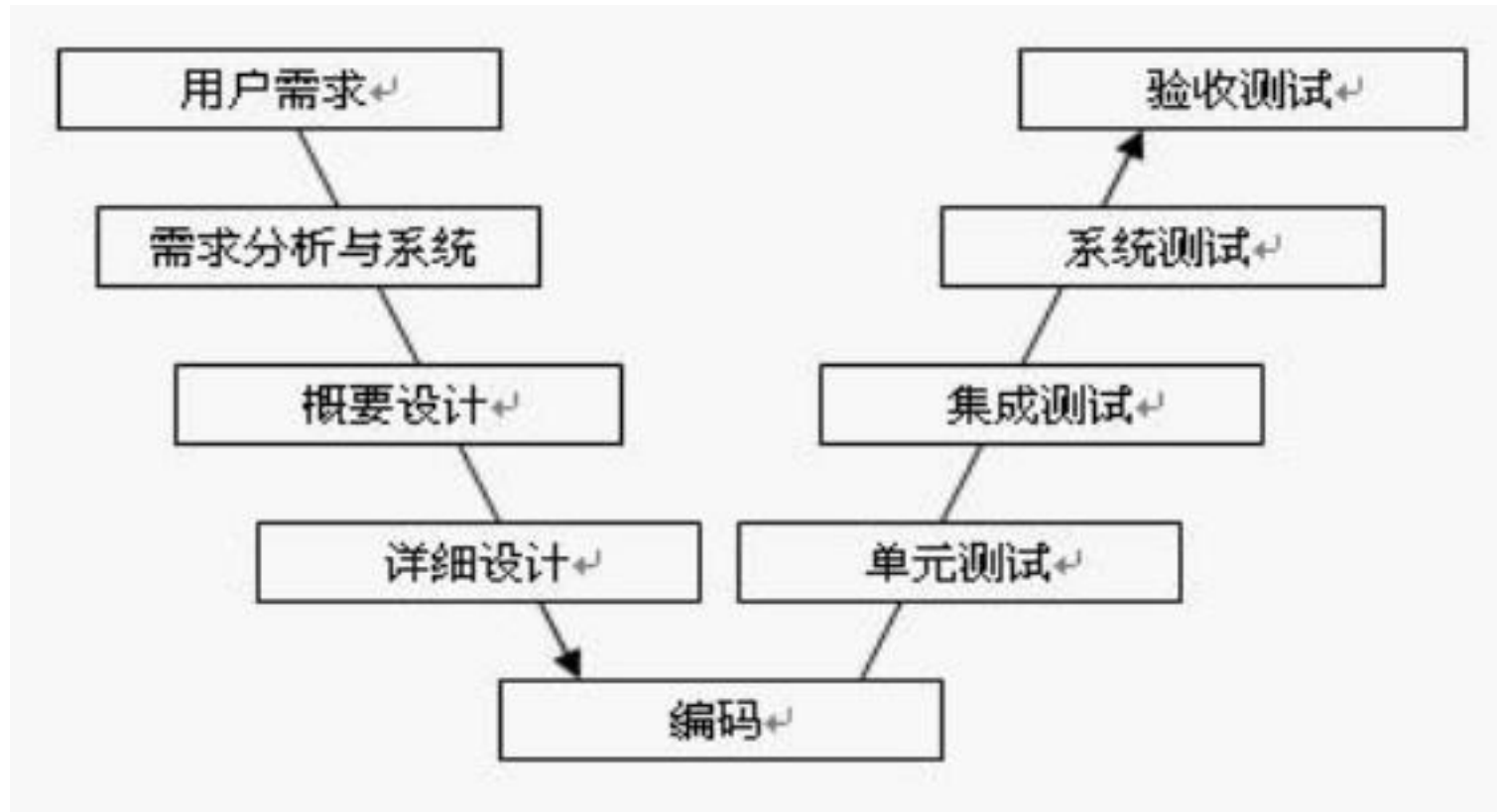


图1-10 软件测试V模型

软件测试过程模型

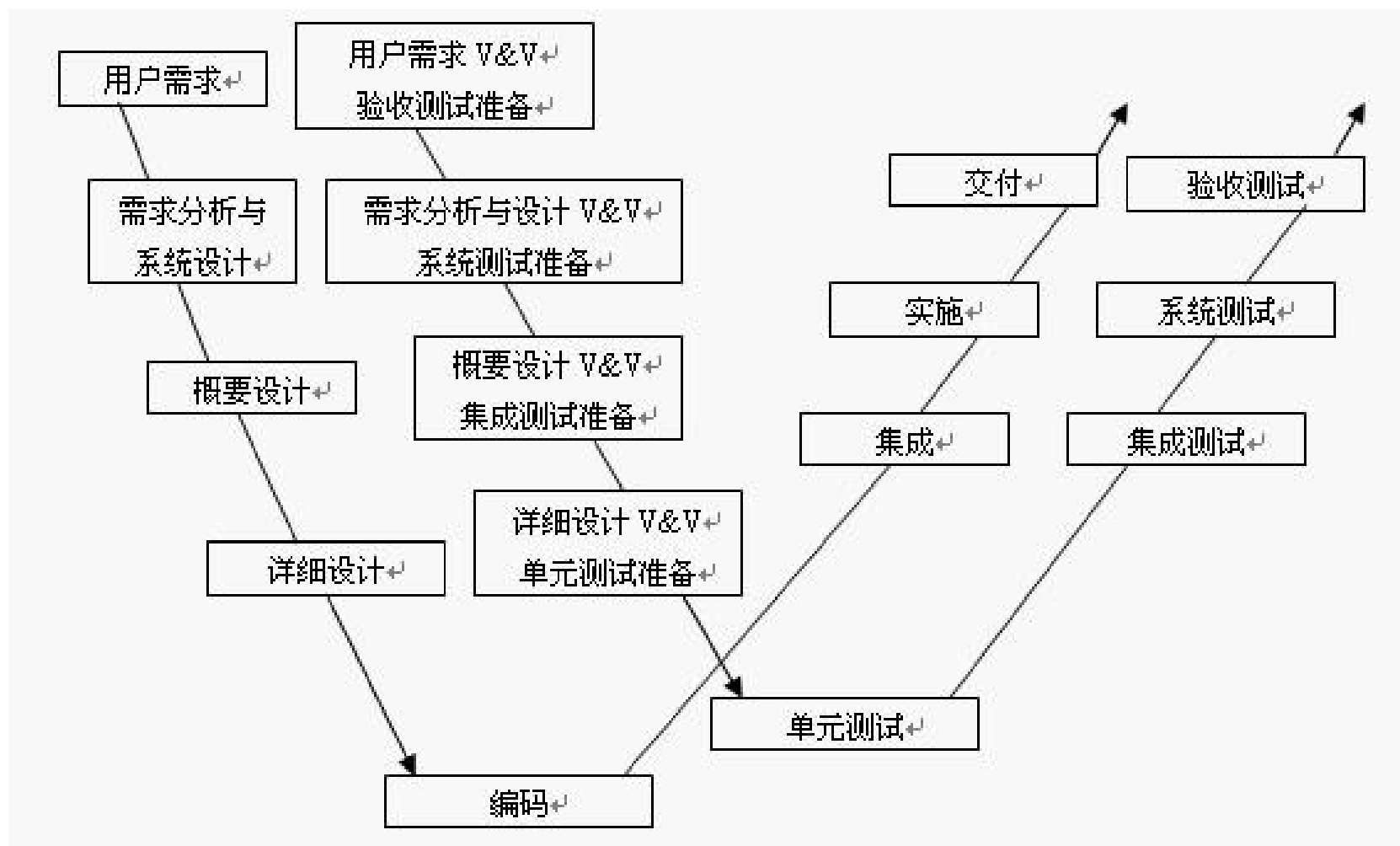


图1-11 软件测试W模型

1.4 软件测试过程

1.4.1 制定测试计划

1.4.2 测试执行过程

1.4.3 软件测试报告

1.4.1 制定测试计划

1、制定测试计划

- 本阶段的主要工作内容
 - 对需求规格说明书的仔细研究
 - 将要测试的产品分解成可独立测试的单元
 - 为每个测试单元确定采用的测试技术
 - 为测试的下一个阶段及其活动制定计划
- 制定计划包括：
 - (1) 概要测试计划
 - (2) 详细测试计划

1.4.1 制定测试计划

2、测试大纲（方案）

测试大纲是软件测试的依据，包括测试项目、测试步骤、测试完成的标准。

- 测试大纲的本质
 - 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。
- 编写测试大纲的好处
 - 保证测试功能不被遗漏，也不被重复测试
 - 合理安排测试人员
 - 使得软件测试不依赖于个人

1.4.2 测试执行过程

1、测试执行过程的三个阶段

(1) 初测期

——测试主要功能和关键的执行路径，排除主要障碍。

(2) 细测期

——依据测试计划和测试大纲、测试用例，逐一测试大大小小的功能、方方面面的特性、性能、用户界面、兼容性、可用性等等；预期可发现大量不同性质、不同严重程度的错误和问题。

(3) 回归测试期

——系统已达到稳定，在一轮测试中发现的错误已十分有限；复查已知错误的纠正情况，确认未引发任何新的错误时，终结回归测试。

1.4.2 测试执行过程

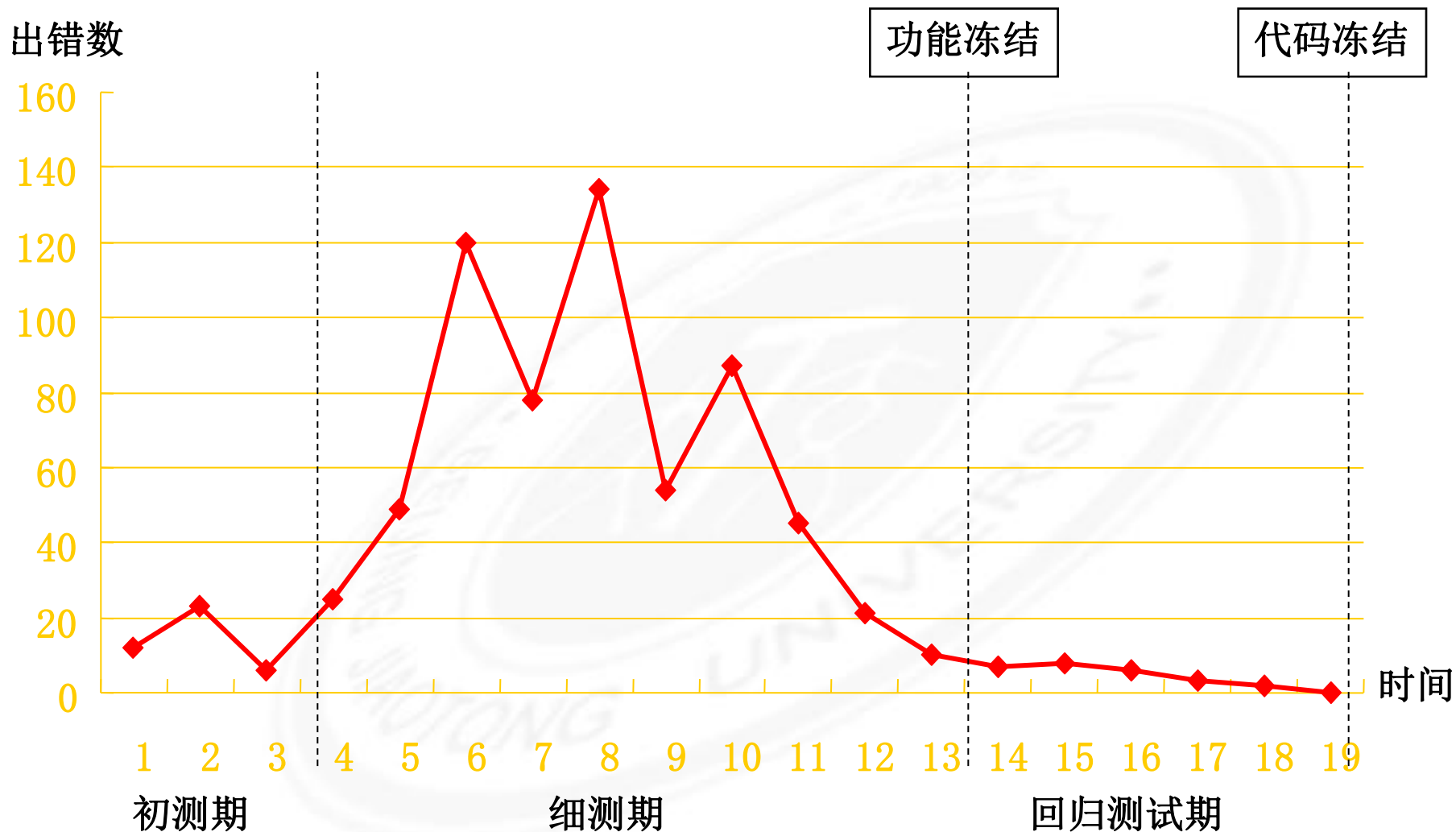


图1-13 三个测试期阶段图示

1.4.2 测试执行过程

2、测试过程中的两个重要里程碑

在测试过程中的两个重要的里程碑是**功能冻结和代码冻结的确定**。这两个里程碑界定出回归测试期的起止界限。

- **功能冻结** (Function/Feature Freeze)
 - 经过测试，符合设计要求，**确认**系统功能和其它特性均不再做任何改变。
- **代码冻结** (Code Freeze)
 - 理论上，在无错误时冻结程序代码，但实际上，代码冻结只标志系统的当前版本的质量**已达到预期的**要求，冻结程序的源代码，不再对其做任何修改。这个里程碑是设置在软件通过最终回归测试之后。

1.4.3 软件测试报告

软件测试报告是软件测试过程中最重要的文档，它的内容包括：

- 记录问题发生的环境：
 - 如：各种资源的配置情况
- 记录问题的再现步骤
- 记录问题性质的说明
- 记录问题的处理进程
 - 问题处理进程从一定角度上反映测试的进程和被测软件的质量状况以及改善过程。

1.5 软件质量保证

1.5.1 软件质量管理

1.5.2 软件能力成熟度模型CMM

1.5.3 ISO9000标准简介

1.5.1 软件质量管理

1、质量管理的概念

- **质量**：是“反映实体满足明确和隐含需要的能力和特性的综合”。因此，质量是一种需要，“是一组固有特性满足要求的程度”。
- **质量管理**：质量管理是指以组织为质量中心、企业全员参与为基础，为追求客户满意和组织所有受益者满意而建立和形成的一整套质量方针、目标和体系。
- 质量管理通过**质量策划**设定组织的质量目标，并规定必要的过程和相关资源；
- 通过**质量控制**监视内部质量过程，排除质量控制过程中可能存在的缺陷隐患；
- 通过**质量改进**提高内部的质量管理能力，改善组织内部的质量过程；通过**质量保证**提供足够的信任证据，表明组织有能力满足客户的质量要求。

1.5.1 软件质量管理

- ④ 质量管理体系：它是质量管理的运作实体，由组织结构、程序、过程、资源四个基本部分组成。
- ④ 质量策划：它是“确定质量以及采用质量管理体系要素和要求的活动”，包括产品策划、质量管理体系管理和运作策划、编制质量计划。
- ④ 质量控制：为达到质量要求所采取的作业技术和活动。质量控制的对象是过程。
- ④ 质量保证：是为了提供足够的信任证据，证明组织有关的各类实体有能力满足质量要求所实施并在必要时进行证实的有计划、有系统的活动。
- ④ 质量改进：是为了向组织的所有受益者提供更多的收益所采用的提高质量过程和效率的各种措施。

1.5.1 软件质量管理

质量管理的发展阶段

(1) **产品质量检验阶段**：这个时期特征是对产品的质量进行检验。产品质量的检验只是一种事后的检查，不能预防不合格品的产生。

(2) **统计质量管理阶段**：它是运用概率论和数理统计的原理，提出控制生产过程，预防不合格产品的思想和方法。即通过小部分样品测试，推测和控制全体产品或工艺过程的质量状况。

(3) **全面质量管理阶段**：从以质量管理专业人员为核心进行质量管理，发展到管理者推动、组织各部门的人员都来进行学习和实行质量管理。

1.5.1 软件质量管理

- ④ 从质量管理理论的发展历史可以看出：
 - 质量管理从单纯的对产品质量进行检验发展到对产品形成过程进行控制。
 - 控制方法从静态发展到动态的、持续的过程改进。
- ④ 质量管理理论发展到今天：
 - 其核心思想已表现为对过程的策划、控制和过程能力的持续改进。

1.5.1 软件质量管理

2、软件质量管理内容

- **软件质量**：是软件产品的特性可以满足用户的功能、性能需求的能力。
- **软件的质量管理**：是软件组织在软件产品生产中的质量策划、质量控制、质量保证和质量改进等与质量有关的相互协调的活动。
- **软件质量管理的内容**包括：
 - (1) 软件质量策划
 - (2) 软件组织的质量过程
 - (3) 软件质量控制与质量保证
 - (4) 软件质量的度量和验证
 - (5) 软件质量改进

1.5.2 软件能力成熟度模型

④ 软件能力成熟度模型 (CMM, Capability Maturity Model):

——是软件行业标准模型，用来定义和评价软件企业开发过程的成熟度，提供如何做才能够提高软件质量的指导。

④ CMM的基本原理:

——CMM将软件组织的过程能力成熟度分为5个级别，每一个级别定义一组过程能力目标，并描述要达到这些目标应该采取的各种实践活动。

④ CMM的主要作用:

——提供了一个软件过程改进的框架。根据CMM模型，软件开发机构或组织能够大幅度的提高按计划、高效率、低成本的提交有质量保证的软件产品的能力。

1.5.2 软件能力成熟度模型

1、CMM的基本过程概念

- 过程：为达到目的而执行的所有的系列步骤。
- 软件过程：开发和维护软件及其相关产品的一组活动、方法、实践和改革。
- 软件过程结构：对组织标准软件过程的一种高级别描述，它描述组织标准软件过程内部的过程元素之间的顺序、接口、内部依赖等关系，以及与外部过程之间的接口和依赖关系。
- 软件过程元素：用于描述软件过程的基本元素，每一个过程元素包含一组定义的、有限的、封闭的相关任务。
- 软件过程定义：CMM中过程定义的基本概念是定义组织的标准软件过程。

1.5.2 软件能力成熟度模型

2、CMM的5个分级标准

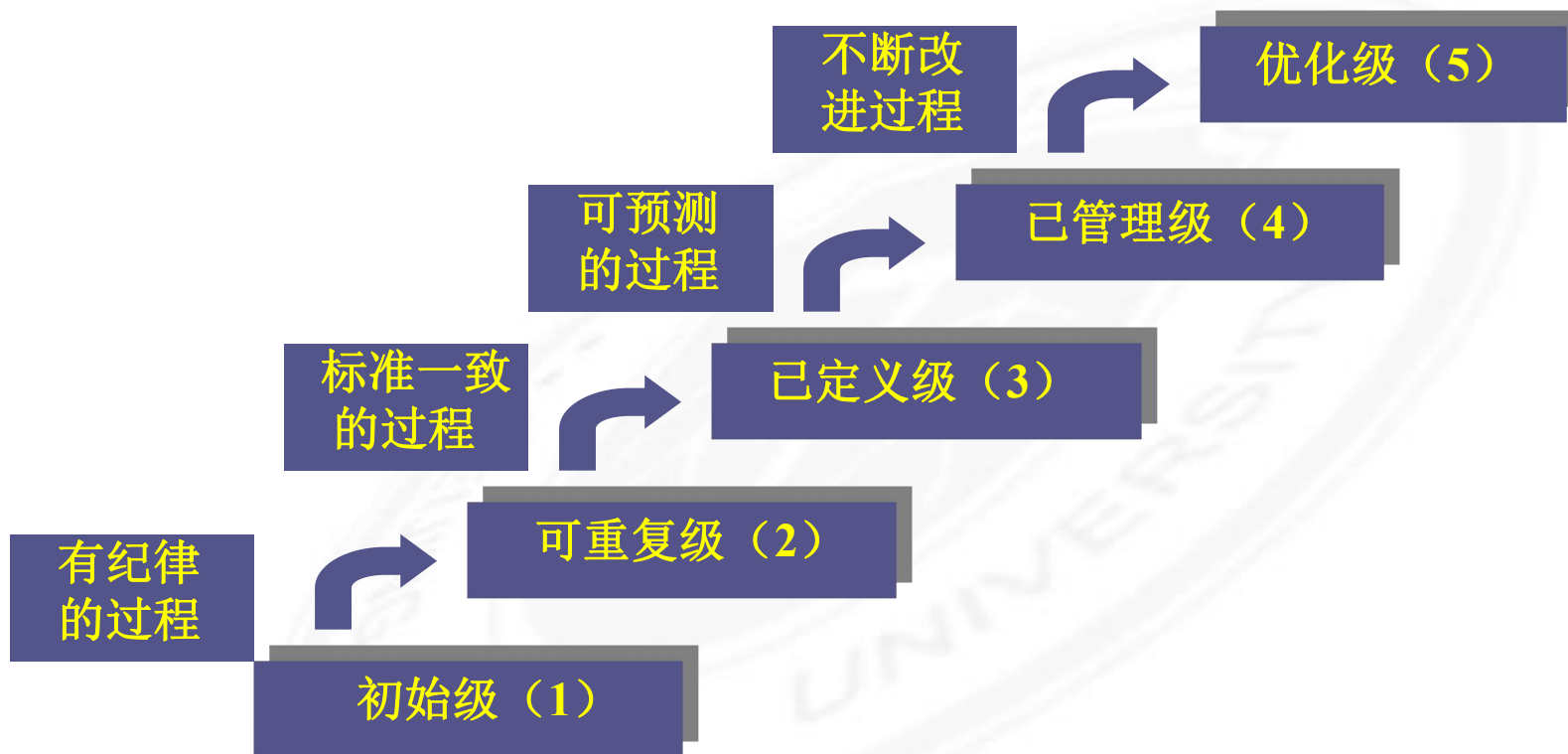


图1-14 软件过程成熟度的5个等级

1.5.2 软件能力成熟度模型

CMM的分级结构和其主要特征：

- **初始级**：其特点是软件过程无秩序，有时甚至是混乱的。
- **可重复级**：已建立了基本的项目管理过程，可用于对成本、进度和功能特性进行跟踪。
- **已定义级**：用于管理的、工程的软件过程均已实现文档化、标准化，并形成了整个软件组织的标准软件过程。
- **管理级**：软件过程和产品质量有详细的度量标准，软件过程和产品质量得到了定量的认证和控制。
- **优化级**：通过对来自过程、新概念和新技术等方面各种有用信息的定量分析，能够不断地、持续性地对过程进行改进。

1.5.2 软件能力成熟度模型

- ④ 除第一级外，CMM的每一级是按照完全相同的内部结构构成的。
- ④ 成熟度等级为顶层，不同的成熟度等级反映了软件组织的软件过程能力和该组织可能实现预期的程度。
- ④ 在每个成熟度级别中（第1级除外），包含了实现这一级目标的若干关键过程域(KPA)。每一级的每个关键过程域进一步包含若干关键实践 (Key Practice, KP)。无论哪一个KPA，其实践都统一按5个公共特性进行组织，即每一个KPA都包含5类KP，使整个软件过程改进工作自上而下形成一种有规律的步骤。

1.5.2 软件能力成熟度模型

- ④ **关键过程域**：是指一系列相互关联的操作活动，这些活动反映了一个软件组织改进软件过程时必须集中力量改进的几个方面。
- ④ **目标**：是指某个关键过程域中的关键实践，它表示每一个关键过程域的范围、边界和意图。
- ④ **公共特性**：为了完成关键过程域中的实践活动，CMM将其活动分为具有**公共特性的5个部分**，包括执行约定、执行能力、实施活动、度量和分析以及验收实施。这些部分的特性有效地指定了一个关键区域的实现范围、结构要求和实施内容。
- ④ **关键实践**：关键实践就是一些主要实践活动。每个关键过程域最终由关键实践所组成，通过实现这些关键实践来达到关键过程域的目标。

1.5.3 ISO 9000 标准简介

④ ISO 9000标准

——最早是为制造行业制定的质量管理和质量评判的一系列标准，后推广至各行业

——定义了一套基本达标的做法，有助于企业能够一致地交付符合客户质量要求的产品（或服务）

④ ISO 9000标准的目标

——在于开发过程，而不是产品

——关心的是进行工作的组织方式，而不是工作成果

④ ISO 9000只决定过程的要求是什么，而不管如何达到。即：ISO 9000指出要做什么，但不指出怎样做。

1.5.3 ISO 9000 标准简介（续）

1、ISO 9000标准的构成

2000版的ISO 9000系列标准主要由下列标准构成：

- ISO9000：2000 《质量管理体系：基础和术语》
- ISO9000：2000 《质量管理体系：要求》
- ISO9000：2000 《质量管理体系：业绩改进指南》

2、ISO 9000标准的要求

3、ISO 9000质量管理的原则

版本后续逐步有升级。

第三方软件测试实验室质量体系

目前主要实施两类质量体系：CNAS和CMA认证。

- **CNAS**：中国合格评定国家认可委员会（China National Accreditation Service for Conformity Assessment, CNAS），由国家认证认可监督管理委员会批准设立并授权的国家认可机构，统一负责对认证机构、实验室和检查机构等相关机构的认可工作。
- **通过CNAS认证**，表示这个实验室获得国家认可，软件测试的品质有保障。

第三方软件测试实验室质量体系

- **CMA**是中国计量认证（可），它表示该实验室的特定检测项目通过了相关部门的专家评审认可，具有一定的权威性。

它是进行某些国内强制检测的实验室必须取得的一种资质，来源于计量法的要求。

- 取得**CMA资质**的实验室可在其通过认可的项目的检测报告上（通常是封面）使用CMA标识。



China National Accreditation Service for Conformity Assessment
LABORATORY ACCREDITATION CERTIFICATE

(Reg)

Software Evaluation

J

(Legal E

No.9 Teaching Building, I

is accredited in ac

Requirements for the

Laboratories(CNAS-CL01

Testing and Calibration L

the service described in t

The scope of accre

bearing the same registr

integral part of this certifi

Effective Date: 2019-02-27

Expiry Date: 2022-12-04

Signed on behalf of China National

China National Accreditation Service for C
Administration of the People's Republic
assessment. CNAS is a signatory of the Int
(ILAC MRA) and the Asia Pacific Laborato
The validity of the certificate can be checked o



中国合格评定国家认可委员会
实验室认可证书

(注册号: CNAS L9487)

兹证明:

北京交通大学软件评测实验室

(法人: 北京交通大学)

北京市海淀区上园村3号第九教学楼, 100044

符合 ISO/IEC 17025: 2017《检测和校准实验室能力的通用要求》
(CNAS-CL01《检测和校准实验室能力认可准则》)的要求, 具备承担本
证书附件所列服务能力, 予以认可。

获认可的能力范围见标有相同认可注册号的证书附件, 证书附件是
本证书组成部分。

生效日期: 2019-02-27

截止日期: 2022-12-04



中国合格评定国家认可委员会授权人

朱华

中国合格评定国家认可委员会(CNAS)经国家认证认可监督管理委员会(CNCA)授权, 负责实施合格评定国家认可制度。
CNAS是国际实验室认可合作组织(ILAC)和亚太实验室认可合作组织(APLAC)的互认协议成员。
本证书的有效性可登陆www.cnas.org.cn获认可的机构名录查询。



资质认定
计量认证证书

证书编号: 2014003535N

名称: 北京交通大学软件评测实验室

地址: 北京市海淀区上园村3号(100044)

经审查, 你机构已具备国家有关法律、行政法规
规定的基本条件和能力, 现予批准, 可以向社会出具
具有证明作用的数据和结果, 特发此证。

检测能力见证书附表。

(你机构对外出具检测报告的法律责任由北京交通大学承担。)

准许使用徽标



发证日期: 2014年11月17日

有效期至: 2017年11月16日

发证机关:



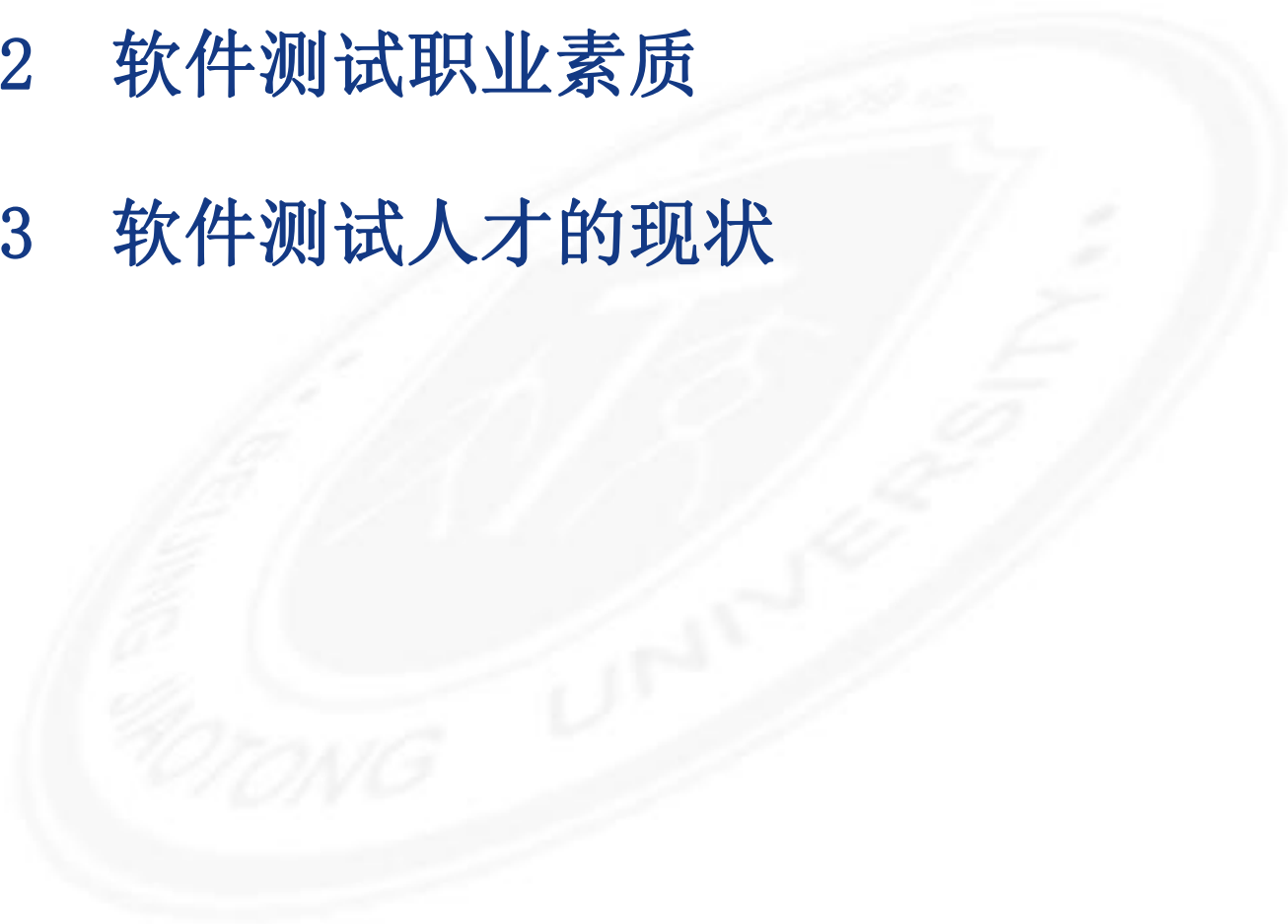
本证书由国家认证认可监督管理委员会制定。在中华人民共和国境内有效。

1.6 软件测试职业

1.6.1 对软件测试的误解

1.6.2 软件测试职业素质

1.6.3 软件测试人才的现状



1.6.1 对软件测试的误解

- ❶ 软件测试技术要求不高，至少比编程容易多了。
- ❷ 软件测试随便找一个能力差的人就能做。
- ❸ 有时间就多测试一些，来不及就少测试一些。
- ❹ 软件测试是测试人员的事，与开发人员无关。
- ❺ 设计—实现—测试，软件测试是开发后期的一个阶段。
- ❻ 如果发布的软件有质量问题，那是软件测试人员的错。

1.6.2 软件测试职业素质

- ④ 软件测试员的工作目标：

——发现潜在的软件缺陷

- ④ 软件测试员应具备的素质：

①具有探索精神

②具有创造性

③坚持不懈精神

④故障排除专家

⑤判断准确

⑥追求完美

⑦沟通能力

1.6.3 软件测试行业现状

软件产业发达的国家：

- 软件测试在人员配备和资金投入方面占据相当的比重。
- 微软为打造Windows2000，1700多个开发人员，以及3200个测试人员，开发和测试人员之比约为三比五。
- HP公司的测试人员和开发人员的比例为一比一，很多先进软件企业的人员配比。

国内：

- 对软件测试不太重视，很多企业没有测试部门，不设置软件测试的岗位，产品质量难以保证。
- 测试人员大都不到开发人员的5% ，随着产业和企业的发展，企业需要大量的测试人员。

1.6.3 软件测试行业现状

软件测试人才紧缺，人才需求快速增长：

- (1) 中国软件产业正在快速增长，需要大量软件相关人才；
- (2) 软件企业的发展要求测试人才达到一个合适的比例。
- 软件测试工程师已成为了亟待补充的关键岗位。
- IBM、微软、三星、西门子、百度、华为、惠普、联想集团等国内外大型IT企业均表现出对成熟软件测试人才的期盼，在全国各大高校招聘测试人员。
- 据统计，国内软件测试工程师的缺口达数十万之多。



本章要点

- ④ 软件测试的背景、软件缺陷和故障的概念
- ④ 软件测试的意义
- ④ 软件开发过程与软件测试的关系
- ④ 软件质量的概念及质量保证体系



本章完