

验证与确认



中国软件评测中心
北京赛迪国软认证有限公司



基本信息（验证）

- 中文名称：验证
- 英文名称：Verification
- 缩写：VER
- CMMI等级：3级
- 过程类型：工程
- 相关过程域： VAL、RD、REQM



基本信息（确认）

- 中文名称：确认
- 英文名称：Validation
- 缩写：VAL
- CMMI等级：3级
- 过程类型：工程
- 相关过程域：RD、TS、 VAL



验证 vs 确认

验证:

指保证软件正确地实现了某个特定要求的一系列活动

确认:

指的是为了保证软件确实满足了用户需求而进行的一系列活动



验证 vs 确认

验证:

You build it right.
确保“你把事情做对了”

确认:

You build the right thing.
确保“你做了对的事”

即“正确的做事”和“做正确的事”

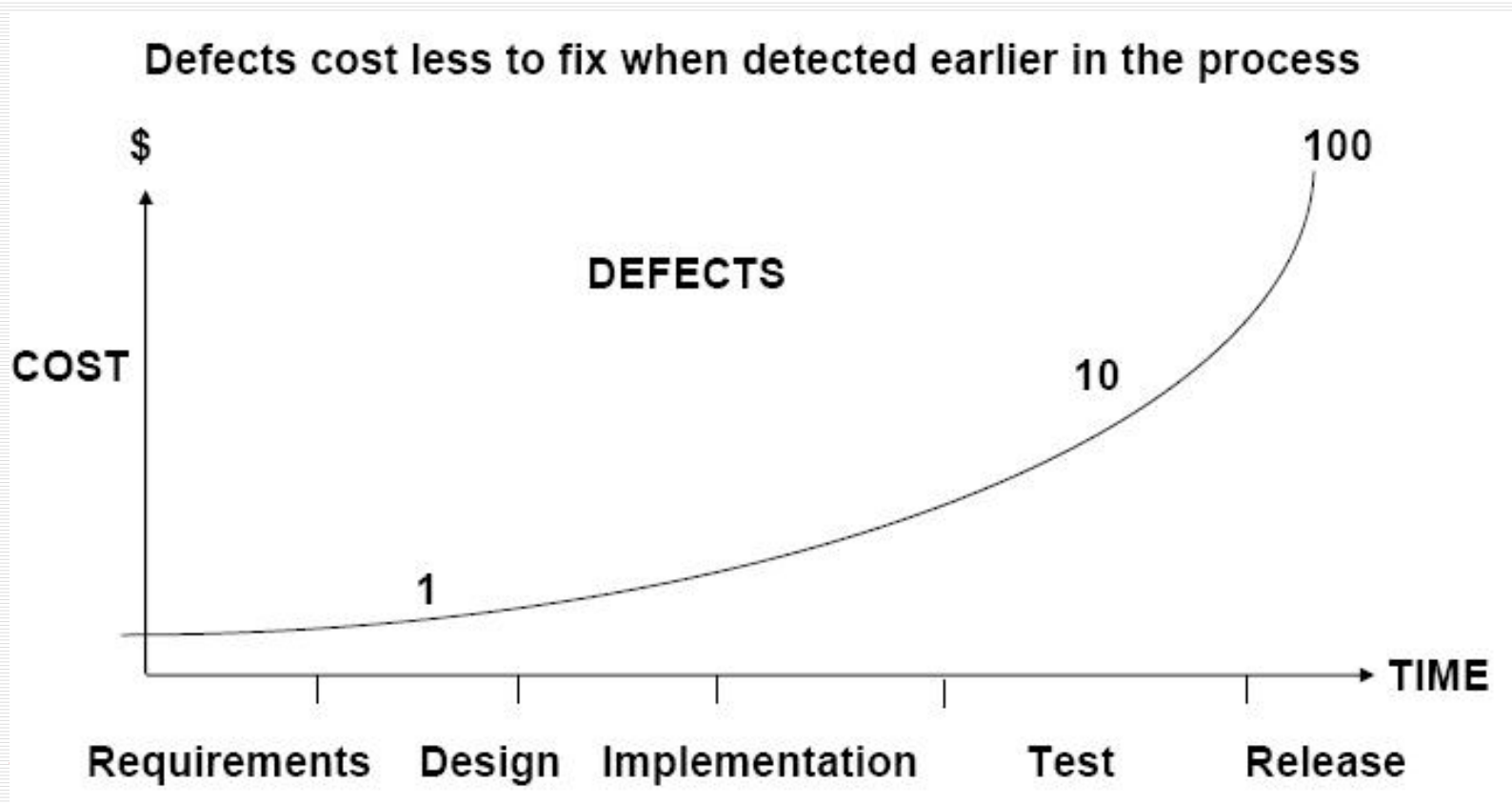


基本概念-什么是评审

- 评审是一种在产品开发过程中尽早发现缺陷的手段
- 评审是一种静态分析，评审对象通常是技术文档、计划、测试用例和测试数据、测试结果等
- 评审是一种很有效的技术，它综合了技术性和管理性措施，手续并不复杂，难度也不很大，所需开支也不高，但效果甚好，专家们已经公认，正规的评审制度对软件的成功是绝对必要的

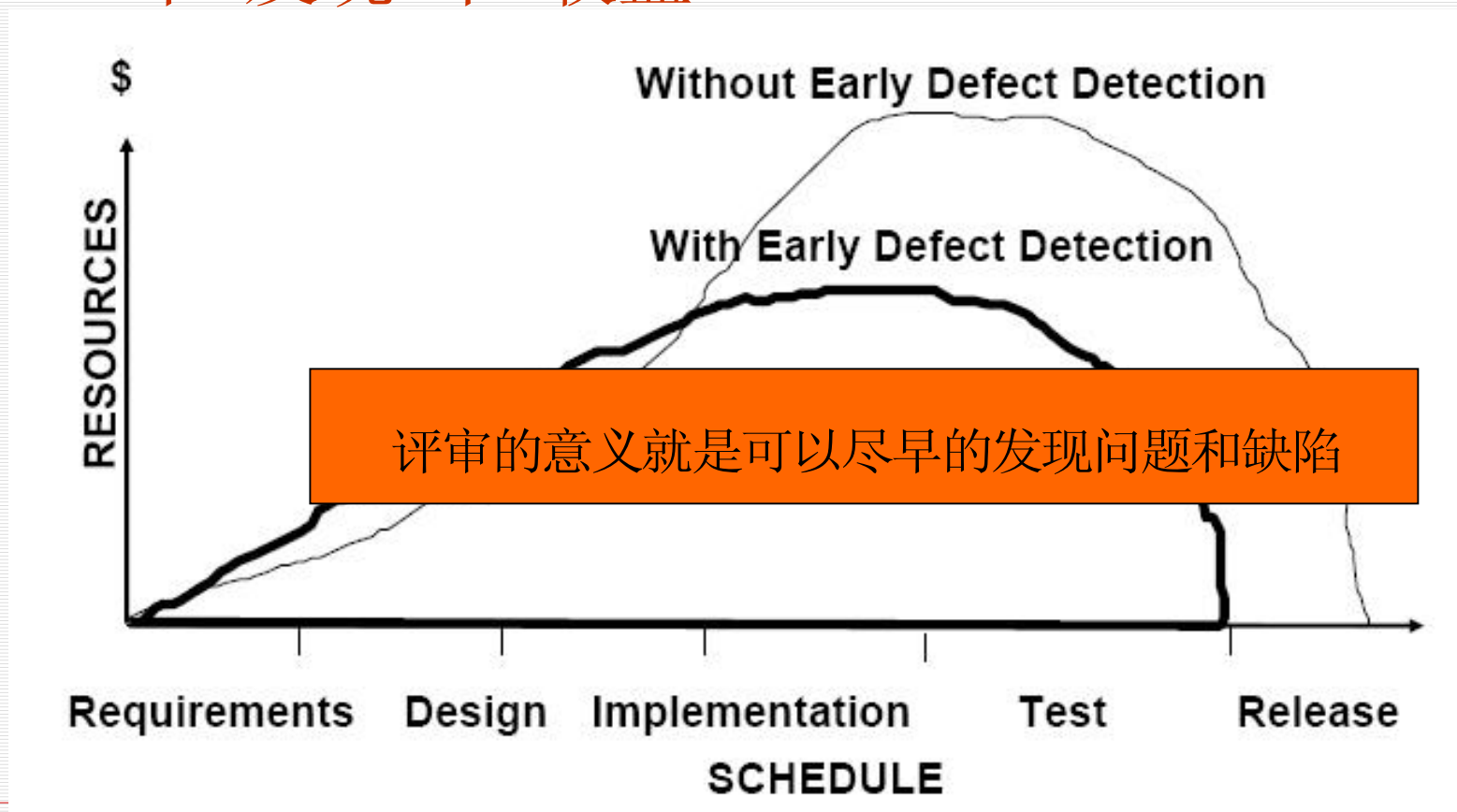
基本概念-评审的意义1

❑ 缺陷纠正的越晚，所耗成本越大



基本概念-评审的意义2

□ “早”发现“早”收益





常用的评审类型

- 正式评审 (Formal Inspection)
- 走查 (Walkthrough)
- 个人评审 (Personal Review)



常用的评审类型-正式评审

□ 特点:

- 规模大, 参与人较多 (与会者业务领域较广)
- 会议角色多, 且分配严格
- 会议前期准备工作较多 (需要预评审等)
- 会议形式正规 (在会议室召开会议等)
- 会后需度量评审数据 (改进, 提高有效性)
- 需定义进入/退出标准

□ 使用场合:

- 被评审的产品较为重要 (需求文档, 用户交付物等)
- 被评审的产品比较成熟



常用的评审类型-走查

□ 特点:

- 人数较少, (与会者业务领域较为相近)
- 形式一般正式
- 角色较少, 定义不太严格

□ 使用场合:

- 专业性较强的产品 (设计文档, 代码评审等)
- 被评审的产品一般成熟
- 正式评审之前的预评审



常用的评审类型-个人评审

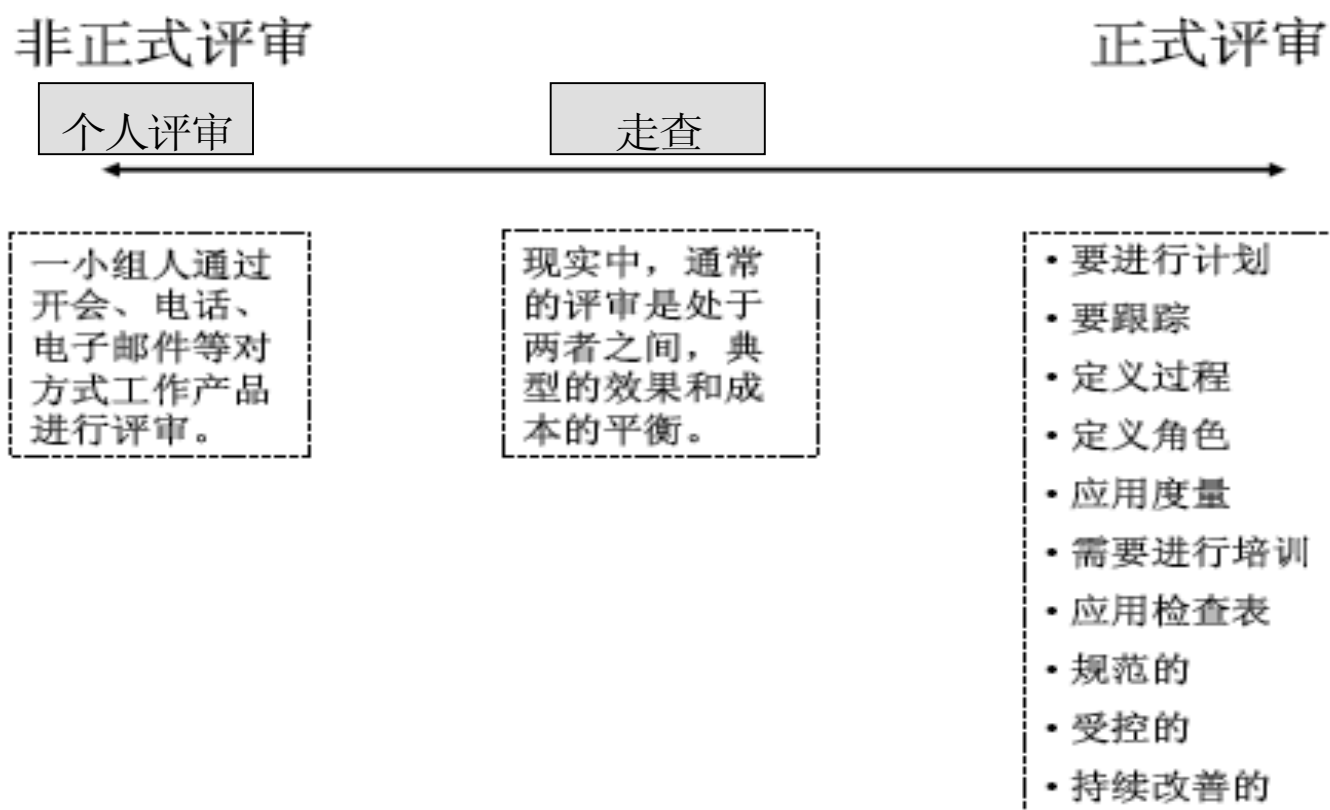
□ 特点:

- 个别人员参与
- 形式多样 (可以非会议形式, 桌查/轮查均可)

□ 使用场合

- 正式评审之前的预评审
- 被评审的产品不成熟也可

评审-总结





常用于评审的文档

- ☐ 需求相关文档 (客户/开发需求...)
- ☐ 项目计划 (开发计划/测试计划/PPQA计划...)
- ☐ 测试用例
- ☐ 测试报告
- ☐ 代码
- ☐ 用户文档 (用户手册/Help/Readme...)



软件测试

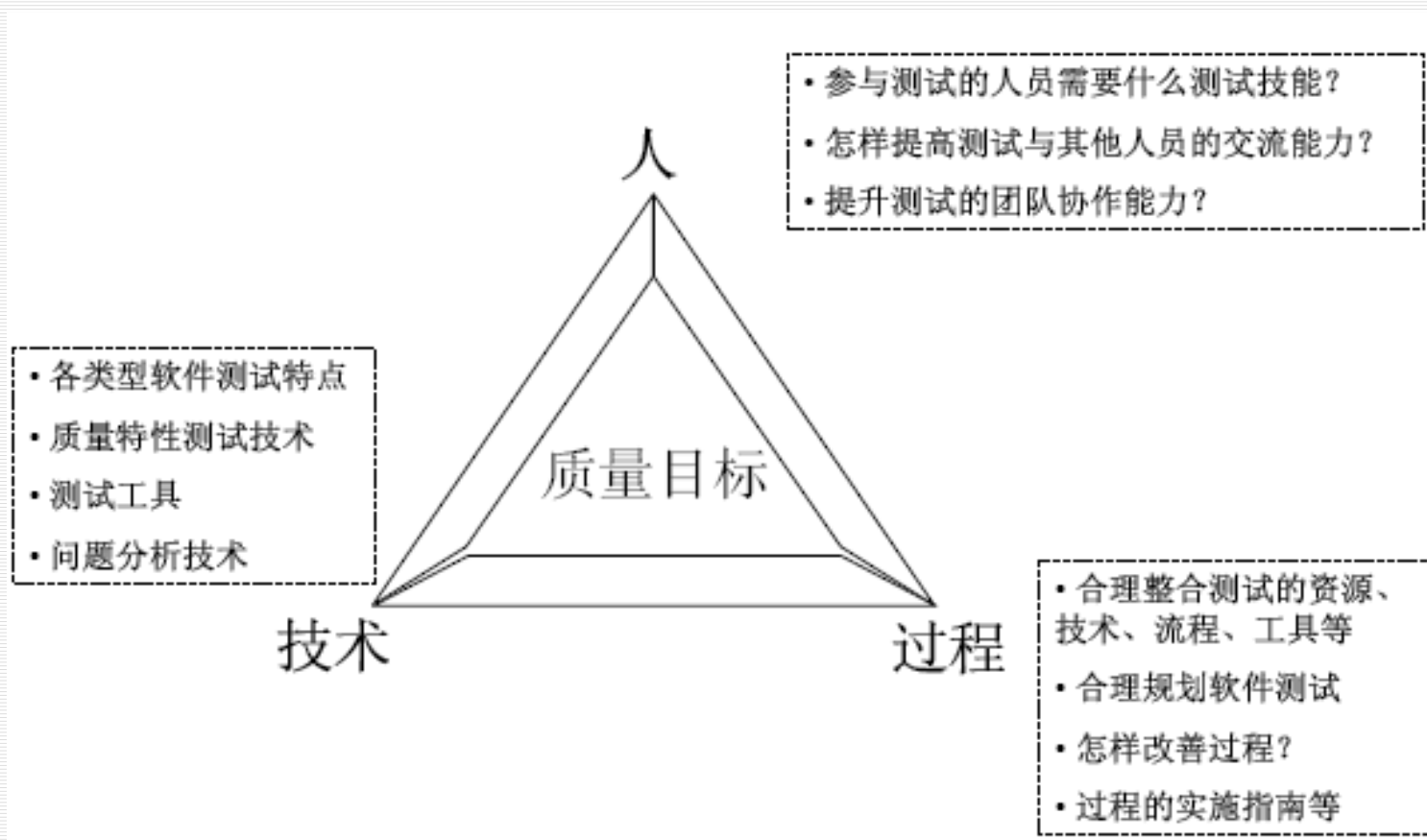
- ☐ 测试的概念
- ☐ 测试需求
- ☐ 测试策略
- ☐ 测试方法
- ☐ 测试过程



测试的概念-常见测试的“问题点”

- ☐ 测试发现不了问题
- ☐ 测试发现问题的时间点太晚
- ☐ 测试发现的是无关紧要的小问题
- ☐ 测试不全面
- ☐ ...

测试的概念-如何改善



测试的概念-如何改善-测试成功的7要素



质量目标(测试要达到什么指标): 项目产品达到的质量要求

测试需求(测些什么): 明确测试对象, 测试内容

测试策略(怎样测): 明确测试阶段, 测试类型

测试技术(怎样实施): 测试用例生成和执行的方法, 测试问题的跟踪和管理方法

测试人员: 负责规划、设计、分析测试的人员, 负责执行测试的人员

测试过程(测试规划与测试过程跟踪): 测试从开始到结束的过程

测试环境(测试软硬件环境)



测试需求

- 测试需求
 - 测试需求体现为软件的测试大纲，是一切测试工作的基础
- 测试需求与项目需求
 - 项目需求通常描述系统能做什么
 - 测试需求既描述系统能做什么，也描述系统不能做什么
- 测试需求的有效性
 - 开发与测试对需求的理解一致
 - 项目需求的可测试性
 - 需求变更的一致性
 - 需求成熟的同步性
 - 场景分析



测试策略-有效性

- 测试策略的有效性
 - 测试内容对需求的覆盖
 - 测试内容的优先等级划分
 - 测试内容重要程度的确定
 - 测试类型的确定
 - 测试阶段的确定

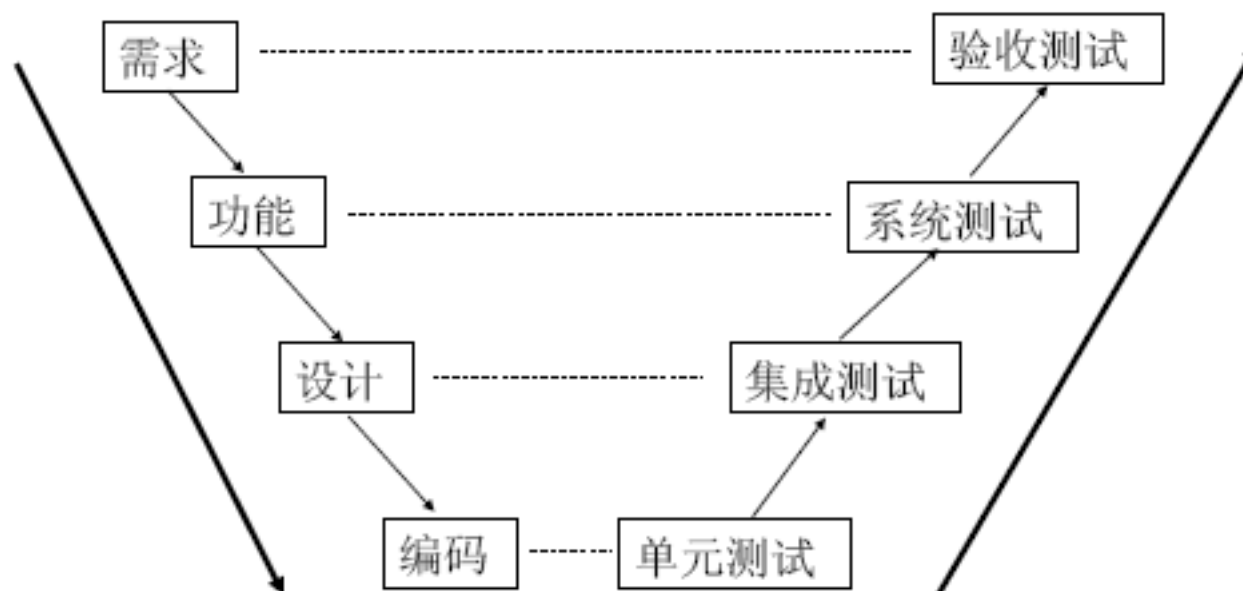


测试策略-软件测试的模型

- ☐ V模型
- ☐ W模型

测试策略-软件测试的模型-V模型

V 模型



V模型反映出了测试活动与分析设计活动的关系。从左到右描述了基本的开发过程和测试行为，非常明确的标注了测试过程中存在的不同类型的测试



测试策略-软件测试的模型-V模型

□ 特点: 较强的针对性

- 单元和集成测试应检测程序的执行是否满足软件设计的要求;
- 系统测试应检测系统功能、性能的质量特性是否达到系统要求的指标;
- 验收测试确定软件的实现是否满足用户需要或合同的要求。

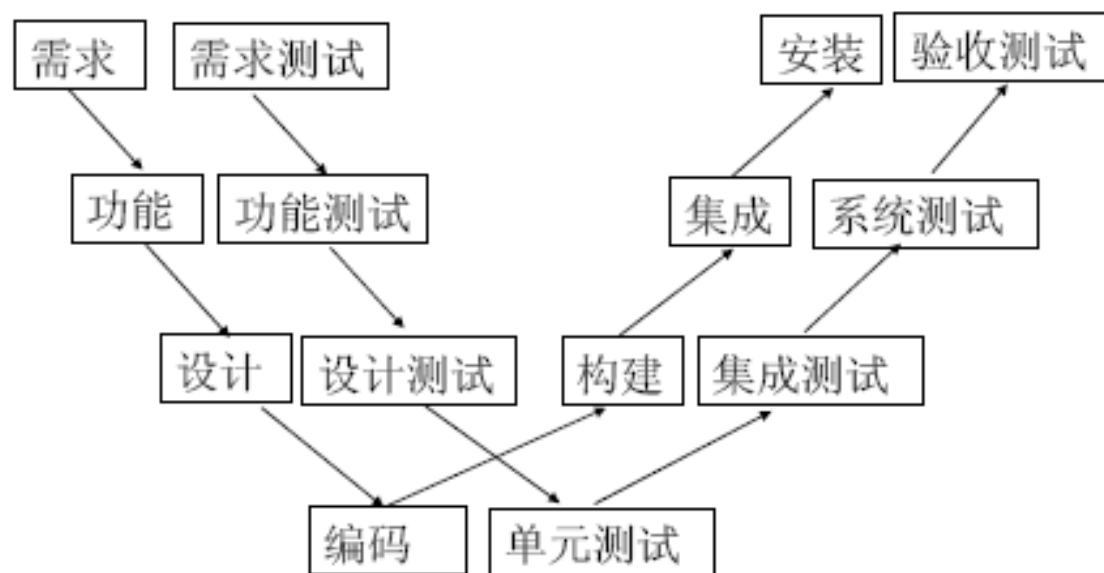
□ 缺点: 测试与开发割裂, 测试参与较晚

- 测试作为在编码之后的一个阶段, 而忽视了测试活动对需求分析、系统设计等活动的验证和确认的功能。

测试策略-软件测试的模型-W模型



W模型



W模型反映出了测试与开发是同步进行的，强调软件各开发阶段中都应同步进行验证和确认活动。而且测试的对象不仅仅是程序，需求、设计等同样要测试。



测试策略-软件测试的模型-W模型

□ 优点

- 测试早参与，可以较深入的了解产品需求；
- 有利于尽早地全面的发现问题；
- 及时全面了解项目难度和测试风险，及早制定应对措施，这将显著减少总体测试时间，加快项目进度

□ 缺点 - 无法支持迭代的开发模型

- 需求、设计、编码等活动被视为串行的，测试和开发活动也保持着一种线性的前后关系

软件测试方法

	测试方法	说明
1	静态测试方法	不执行被测程序的情况下扫描源程序，检查程序模块间的接口是否一致及变量的定义和使用情况。通过对被测试软件的源程序的控制流(如函数调用关系、函数的控制流程图)和数据流(如外部变量使用情况等)的分析来检验程序。
2	动态测试方法	在程序执行的过程中跟踪程序的运行情况，得到程序运行的覆盖率、数据流的变化和程序的执行动态行为。
3	“白盒子”测试方法	检查软件的内部活动是否符合设计要求的方法。它检查软件内部的逻辑结构，是以仔细检查过程的细节为基础，通过提供一组指定条件和循环的测试用例，对穿过软件的逻辑路径进行测试，可以在不同点检查程序的状态，以确定实际状态与预期状态是否一致。白盒子测试也称为结构测试。
4	“灰盒子”测试方法	测试系统部件之间的接口。
5	“黑盒子”测试方法	着眼于软件的外部特性，而不考虑软件的内部逻辑结构。黑盒子测试指在软件的接口上进行测试，即看它能否满足功能要求，输入能否被正确地接收，并正确地输出结果，以及能否保持外部信息的完整性等。黑盒子测试也称为功能测试。



测试过程-理念

□ 尽早测试

- 测试人员尽早参与软件项目，及时开展测试的准备工作
- 尽早的开展测试执行工作 (单元测试)

□ 全面测试

- 对软件的所有产品进行全面的测试，包括需求、设计文档，代码，用户文档等等
- 软件开发及测试人员(有时包括用户)全面的参与测试工作 (需求的验证和确认活动)

□ 全过程测试

- 测试人员要充分关注开发过程，对开发过程的各种变化及时做出响应
- 测试人员要对测试的全过程进行全程的跟踪

□ 独立的、迭代的测试

- “独立的、迭代的测试”着重强调了测试的就绪点，也就是说，只要测试条件成熟，测试准备活动完成，测试的执行活动就可以开展



测试过程-步骤

□ 计划测试

- 估计测试范围、规模、工作量、进度和成本
- 制定明确的测试进入推出标准
- 设计测试用例
- 测试环境
 - * 是否与开发环境一致
 - * 是否与用户环境一致

□ 测试执行

- 按照测试计划执行测试，记录发现的不具合点 (Bug)
- 变更的管理

测试计划

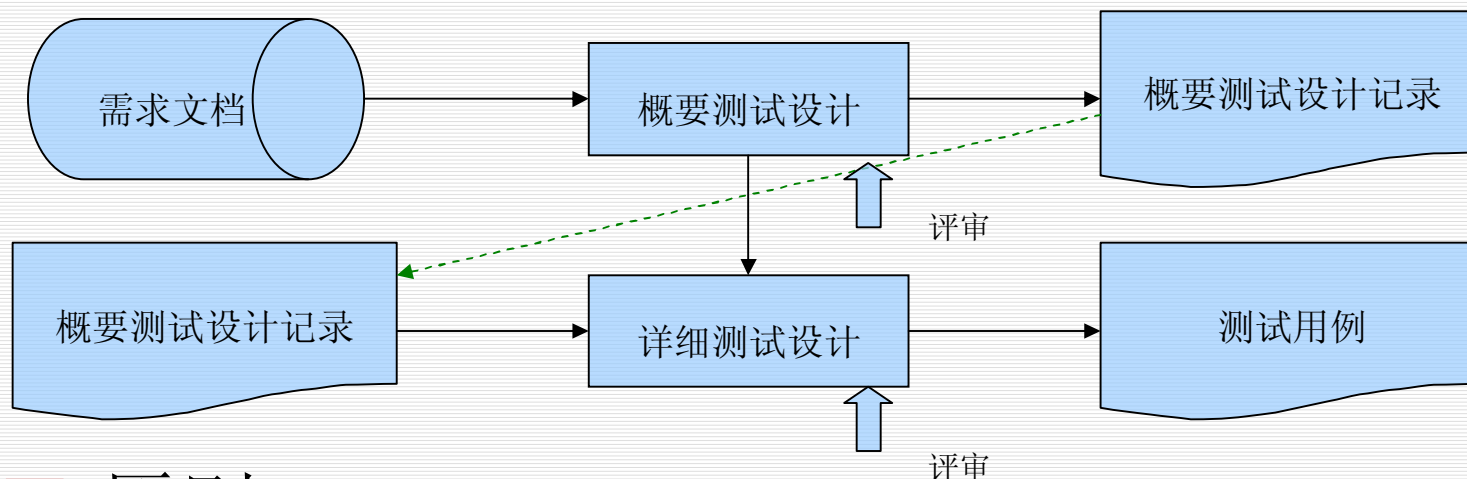
□ 分析测试结果及测试改进

中国软件评测中心 / 北京赛迪国软认证有限公司

测试过程-如何设计测试用例(1/2)



□ 步骤



□ 原则

- 覆盖全面 (合法/非法, 边界值/越界值, 极限值等)
- 测试结果可判断 (包含期望测试结果)
- 测定结果可再现 (同样的操作, 同样的结果)
- 反映测试策略 (优先级定义)

测试用例



测试过程如何设计测试用例(2/2)

□ 方法

- 等价类划分法：有效等价类/无效等价类
- 边界值分析法：数字边界值/字符边界值，空白/零
- 因果图法：分析输入/输出之间的因果关系，设计不用的输入组合，形成组合测试用例
- 错误推测法：依赖经验、直觉，给出可能存在缺陷的条件

□ 难点

- 设计粒度太粗：深入程度不够，有漏测/无效测试的风险
- 设计粒度太细：过于具体，自我发挥空间较少，不利于发现更多的潜在问题
- 设计粒度影响测试用例的复用



测试过程-测试用例相关误区

- ❑ 测试重执行，轻设计；
- ❑ 测试设计未进行评审；
- ❑ 测试输入和规程过于详细、过于精确；
- ❑ 严格按照测试用例执行，不关心其他现象
- ❑ 只检查产品应该做什么，而不检查其不应该做什么
- ❑ 测试用例描述不清
- ❑ 缺陷报告机制缺乏
- ❑ 客户问题的处理缺乏跟踪



测试过程-测试数据分析

- 测试对象数据分析
 - 测试对象的质量现状
 - 测试对象的质量发展趋势
 - 测试对象的质量预测
- 测试自身数据分析
 - 测试的有效性(驳回率，发现问题/验证问题的效率等)



测试过程-测试数据分析方法

技术	应用对象	应用场景
分布图	测试对象 测试自身	用于长期测试任务积累，主要分析测试过程性能的宏观趋势，结果是形成一定的范围划分。
趋势图	测试对象 测试自身	用于分析项目产品质量的发展趋势。趋势图看的是趋势，并不是具体采集点的差异。可用于项目测试的全过程。
柱状图	测试自身	展示项目或过程中的度量分布，主要适合等量数据和连续数据的表示，比如测试过程的工作量投入。
条状图	测试对象	测试对象的质量状况；测试对象质量状况反映出的开发状态。
排列图	测试对象 测试自身	用于找寻项目或过程中可能存在的“80/20”定律，找出项目的主要关注区域或影响因素
控制图	测试自身	基于统计技术，用于测试过程管理。

测试报告



测试过程-管理

□ 策划测试过程

□ 把握需求

- 测试人员早期参与需求的获取和分析中，有助于加深测试人员对需求的把握和理解。

□ 变更控制

- 依据“全过程测试”理念，测试组密切关注开发过程，跟随进度计划的变更调整测试策略，依据需求的变更及时补充和完善测试用例。

□ 度量与分析

- 对测试过程的度量有利于及时把握项目情况，对过程数据进行分析，很容易发现优势劣势，找出需要改进的地方，及时调整测试策略。

中国软件评测中心 / 北京赛迪国软认证有限公司

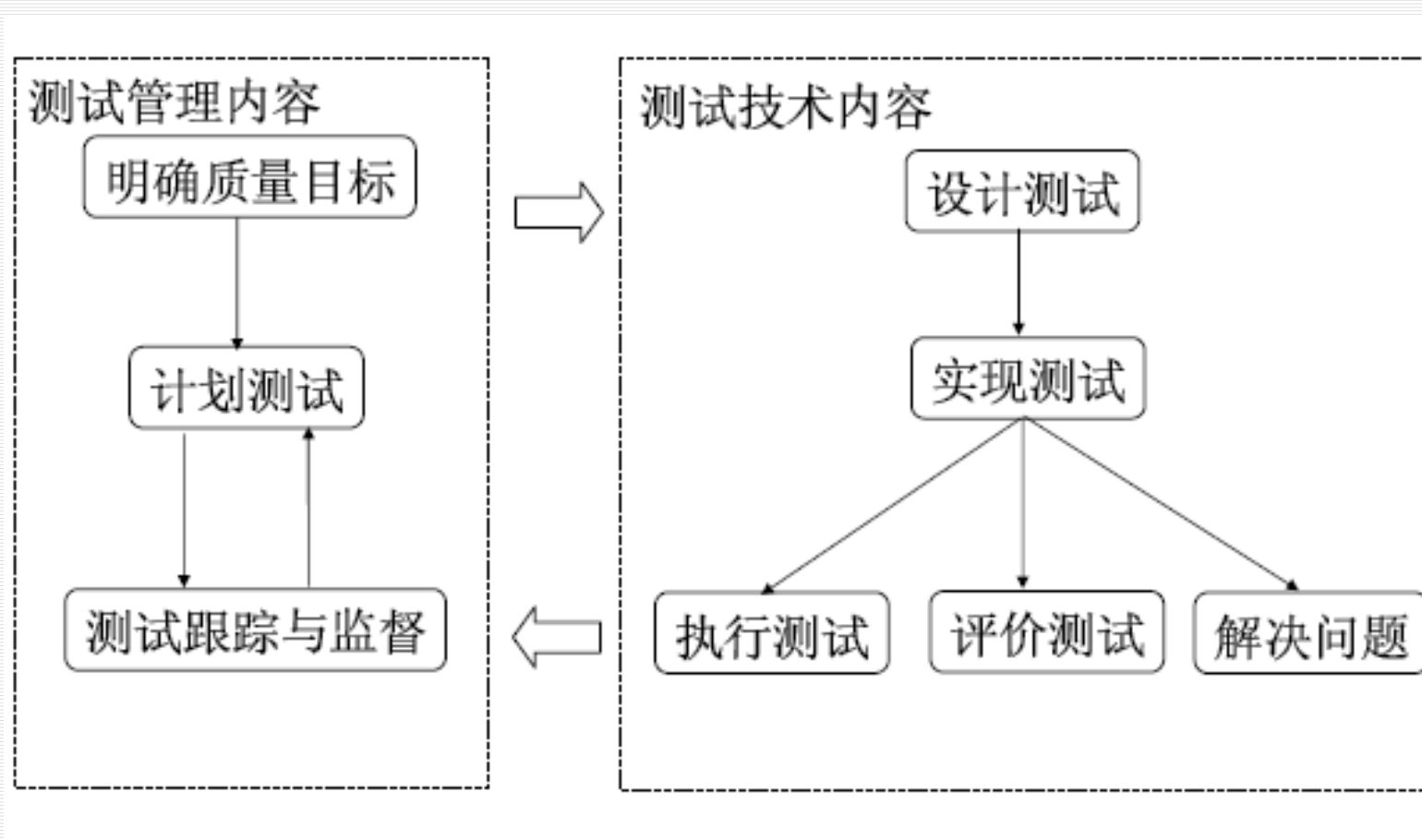


测试过程-一些误区

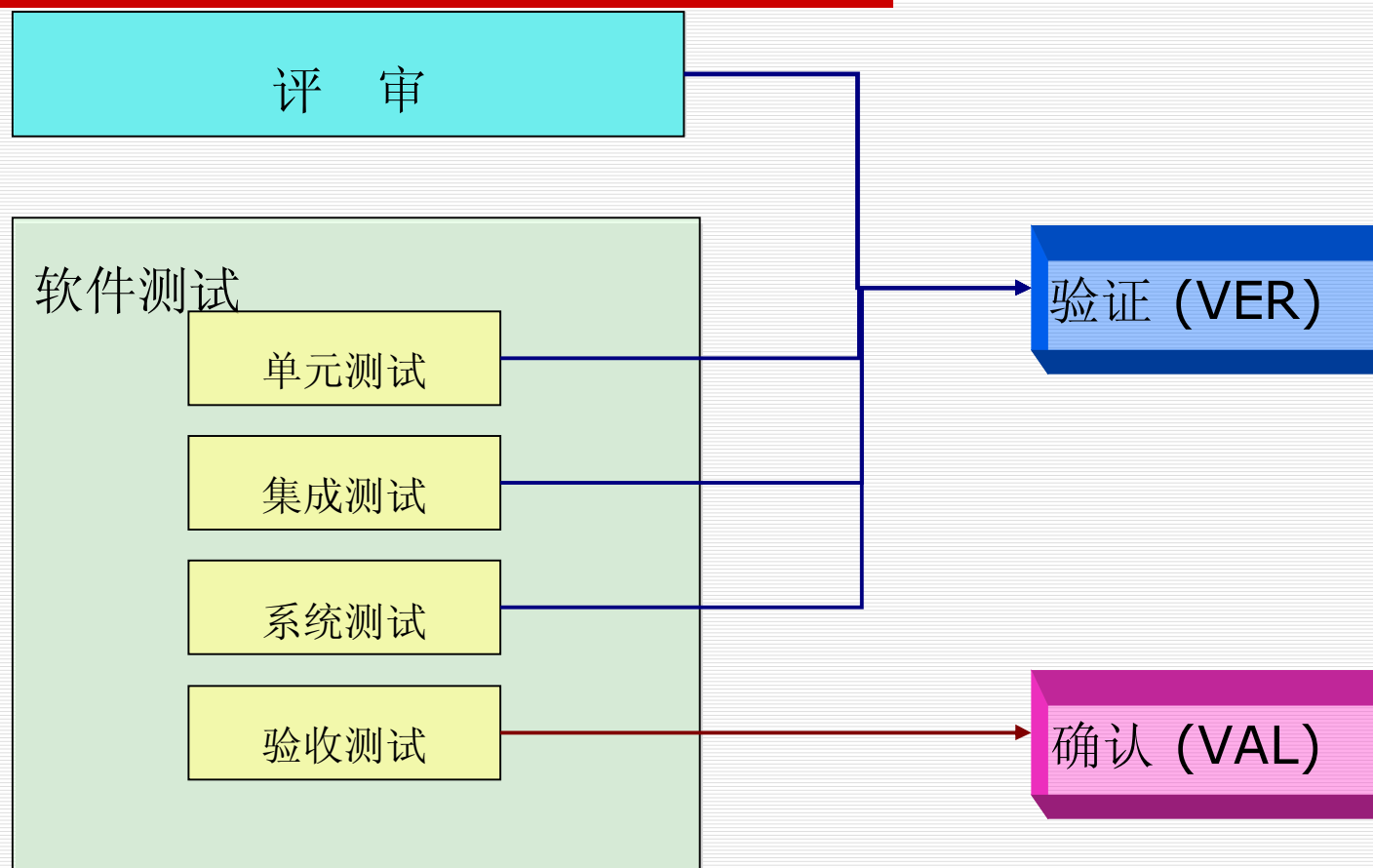
- 测试工作作为所有程序员新手的工作
- 从低绩效开发人员中招聘测试人员
- 测试人员不需要是领域专家
- 不从客户服务人员或技术支持人员中寻求测试后选人
- 坚持测试人员要能够编程
- 测试队伍缺乏多样性
- 开发人员和测试人员物理上分开
- 相信程序员不能测试他们自己的代码
- 程序员较少进行测试的培训和鼓励进行测试

总结

软件测试过程



总结





验证

目的：

确保选定的工作产品符合其指定的需求。

注释：

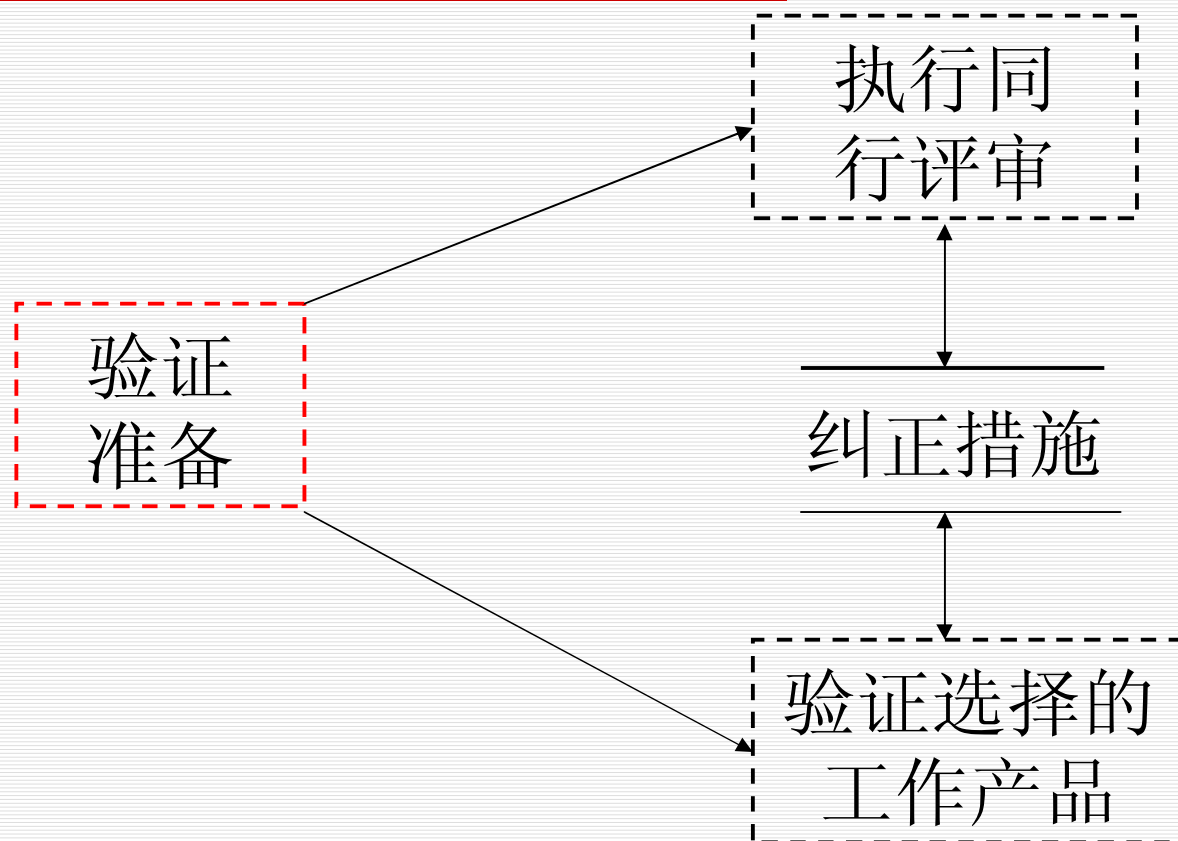
- 验证过程域包括：验证准备、验证执行及纠正措施识别。
- 验证包括产品及中间工作产品的验证，将其与选定的客户需求、产品需求及产品组件需求加以比较。
- 验证是一种渐进式过程因为它发生于产品及工作产品的开发过程中，从需求验证开始，经工作产品到最终完成产品的验证。
- 验证工作产品可以增加产品符合客户需求、产品需求及产品组件需求的可能性。



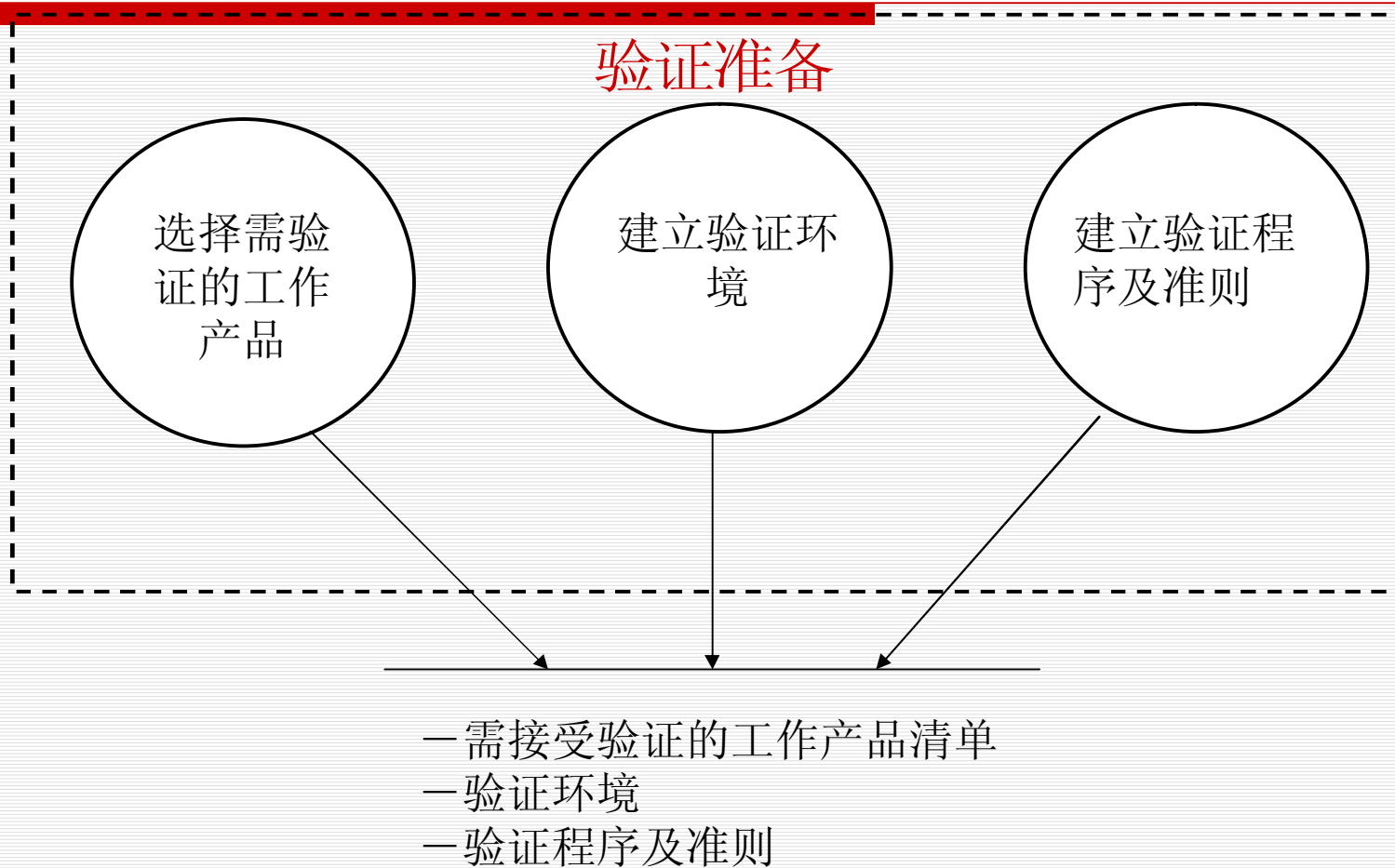
验证——特定目标及特定实践

- SG 1 验证准备
 - SP 1.1 选择需验证的工作产品
 - SP 1.2 建立验证环境
 - SP 1.3 建立验证程序及准则
 - SG 2 执行同行评审
 - SP 2.1 准备同行评审
 - SP 2.2 进行同行评审
 - SP 2.3 分析同行评审资料
 - SG 3 验证工作产品
 - SP 3.1 执行验证
 - SP 3.2 分析验证结果
-

验证



验证准备



如何实施验证-SG1 准备验证



- 如何选择需要评审的工作产品

1. 根据项目需求和目标

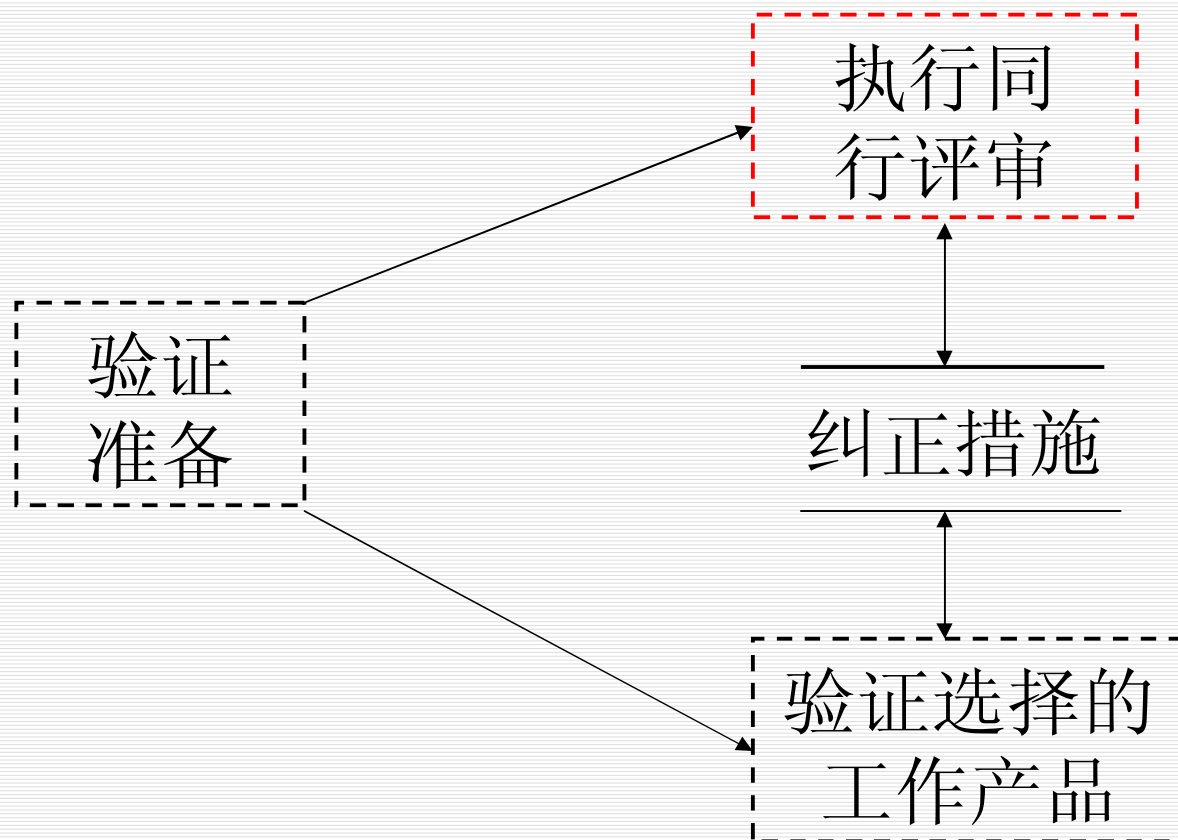
- 关注客户满意度：选择交付给客户的文档；
- 关注客户需求的捕获：选择需求相关文档
- ...

2. 根据项目风险

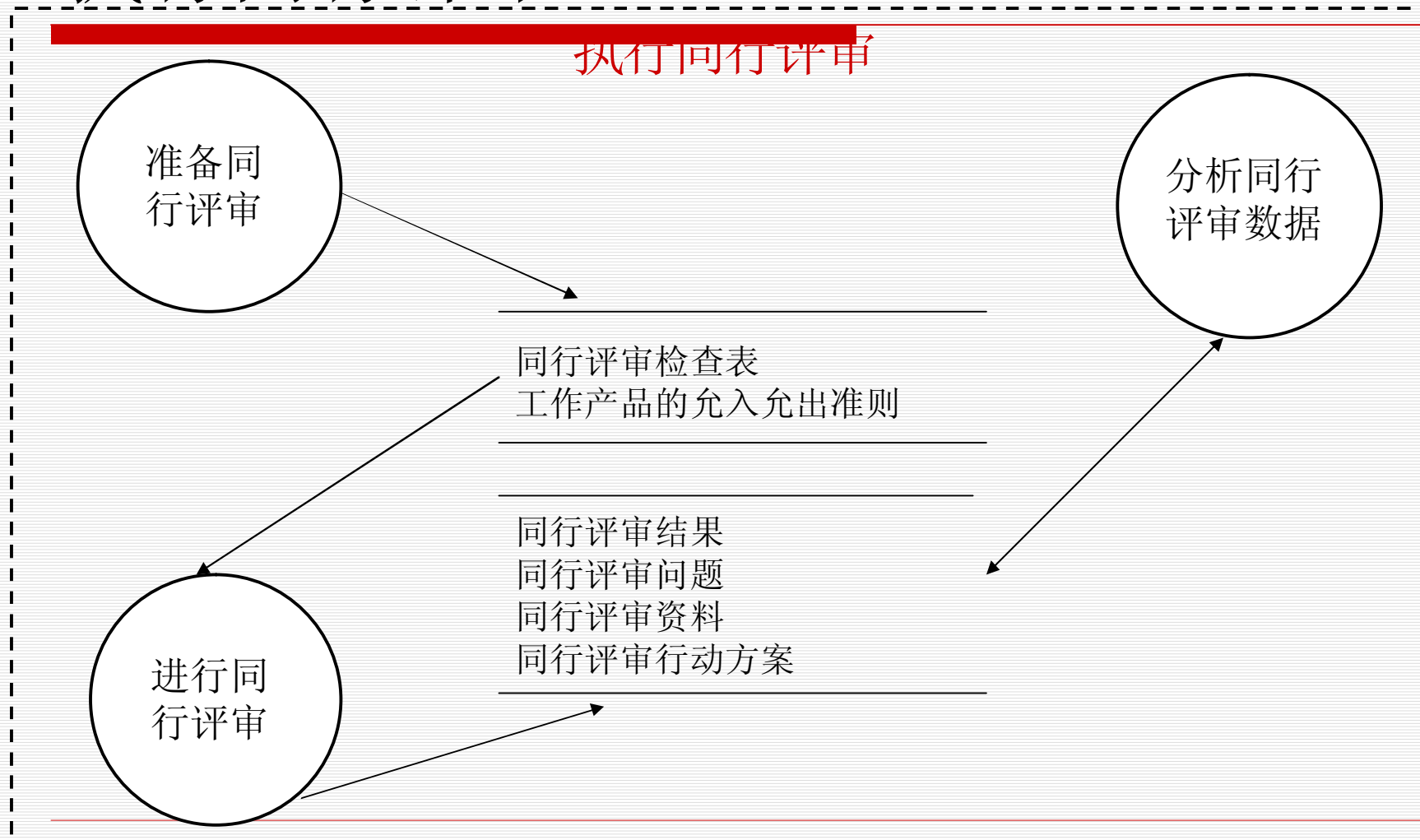
- 技术实现风险：选择相关技术文档，代码等
- 设计风险：选择相关技术文档，代码等
- ...

明确各个需要评审的工作产品的重要程度

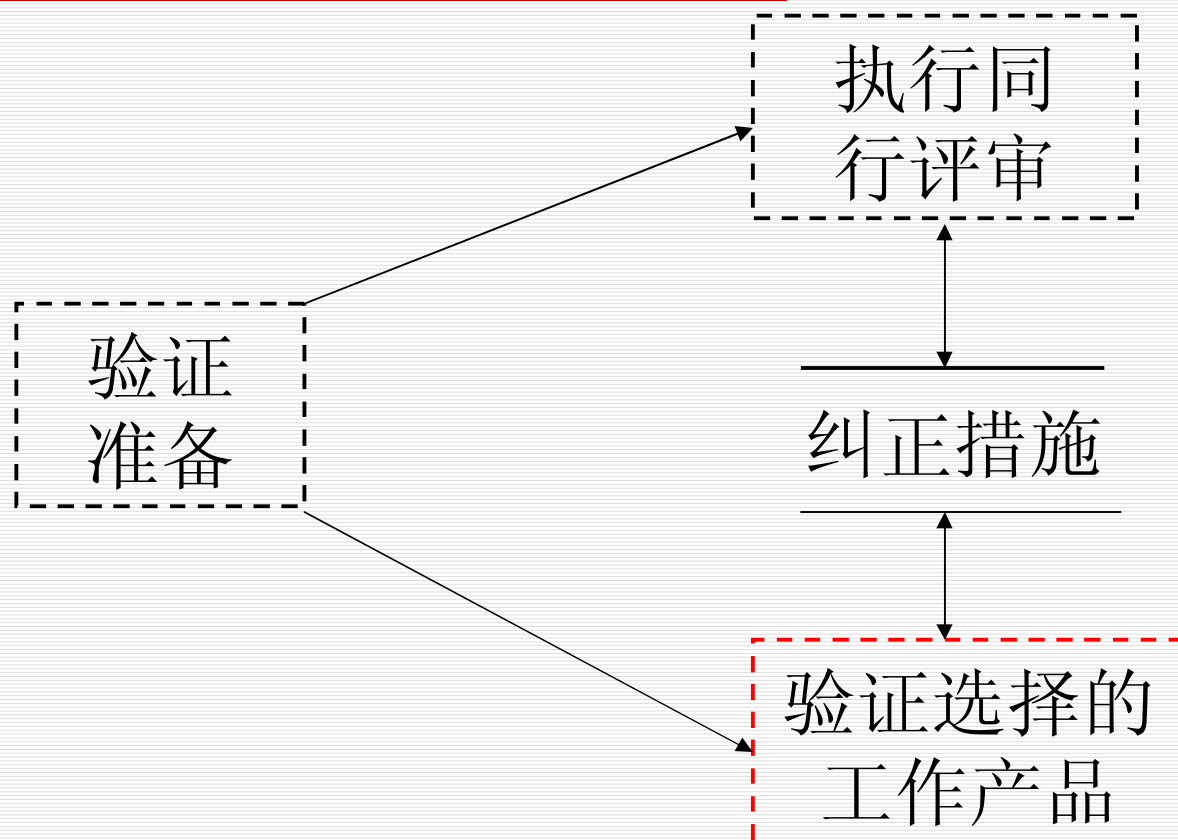
验证



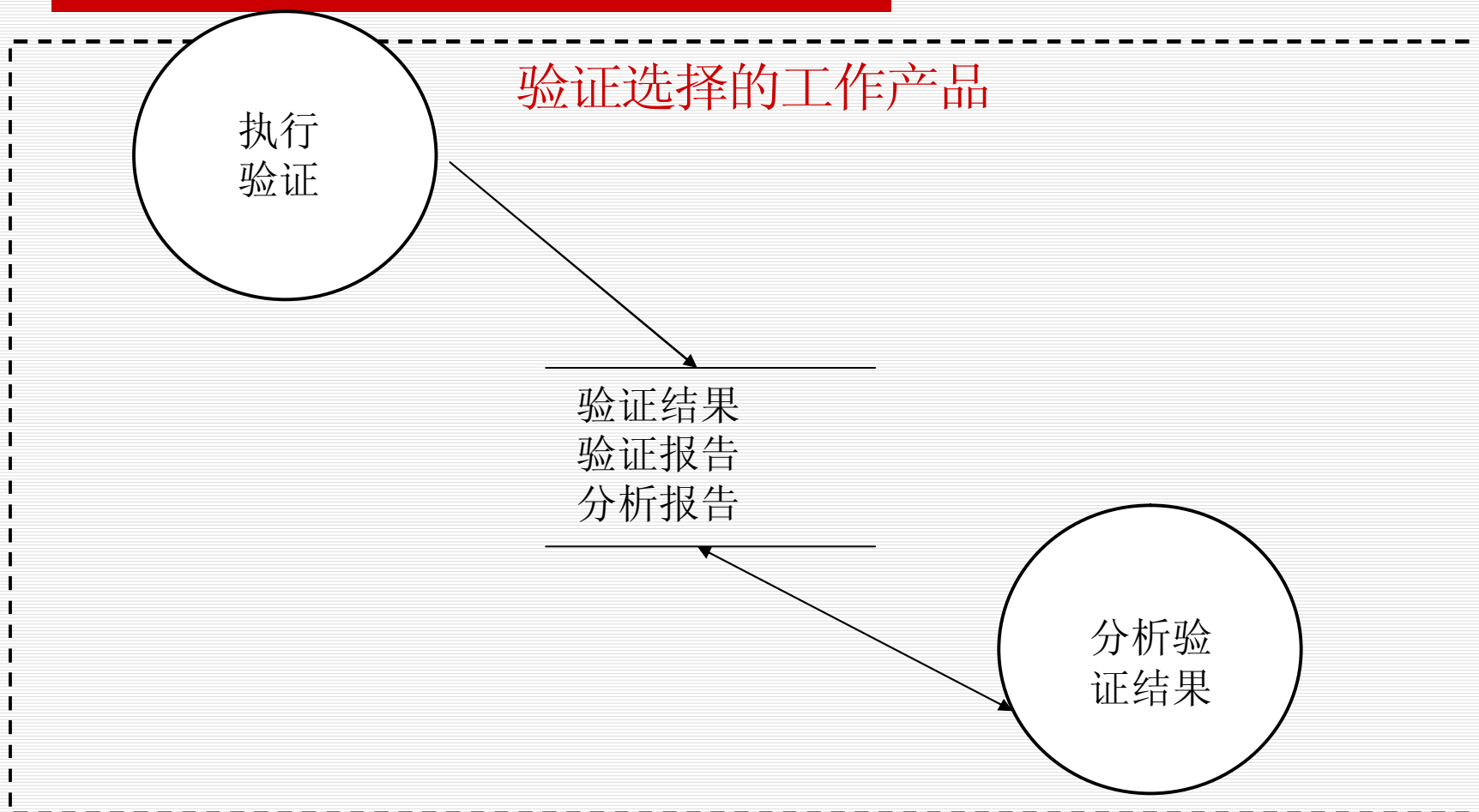
执行同行评审



验证



验证选择的工作产品





验证没做好？

- 如果验证没做好，有什么现象？
 - 应当在早期发现的缺陷都留到了生命周期的晚期处理
 - 测试的产品不能达到需求或设计期望
 - 增加了产品集成或测试时间
 - 对于哪些组件需要做验证、做哪些验证，在技术人员之间存在不同认识



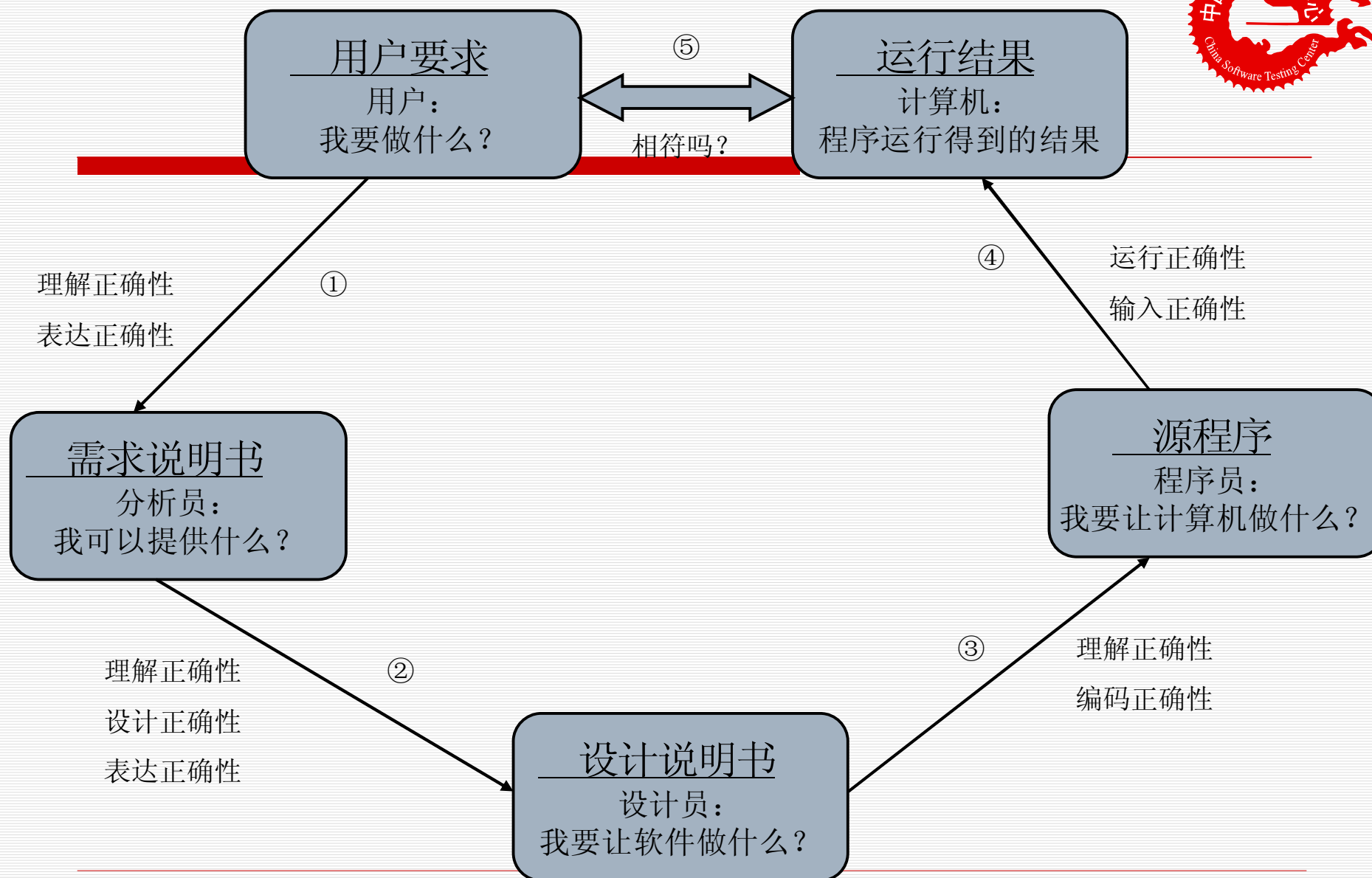
VER---总结

□ 测试

- 测试方案（或测试计划）
- 测试用例
- BUG跟踪记录
- 测试报告

□ 同行评审

- 同行评审计划
- 同行评审记录（预读记录、同行评审报告等）
- 同行评审问题跟踪



正确的用户需求

需求分析

设计

正确的需求

错误的需求

正确的设计

错误的设计

基于错误需求的设计

实现

正确的程序

错误的程序

基于错误设计的程序

基于错误需求的程序

测试

正确的功能

可定位的错误

无法定位的错误

隐藏的缺陷



确认

目的:

证明置于预期环境中的产品或产品组件，
可满足其预期的使用需求。



确认——特定目标及特定实践

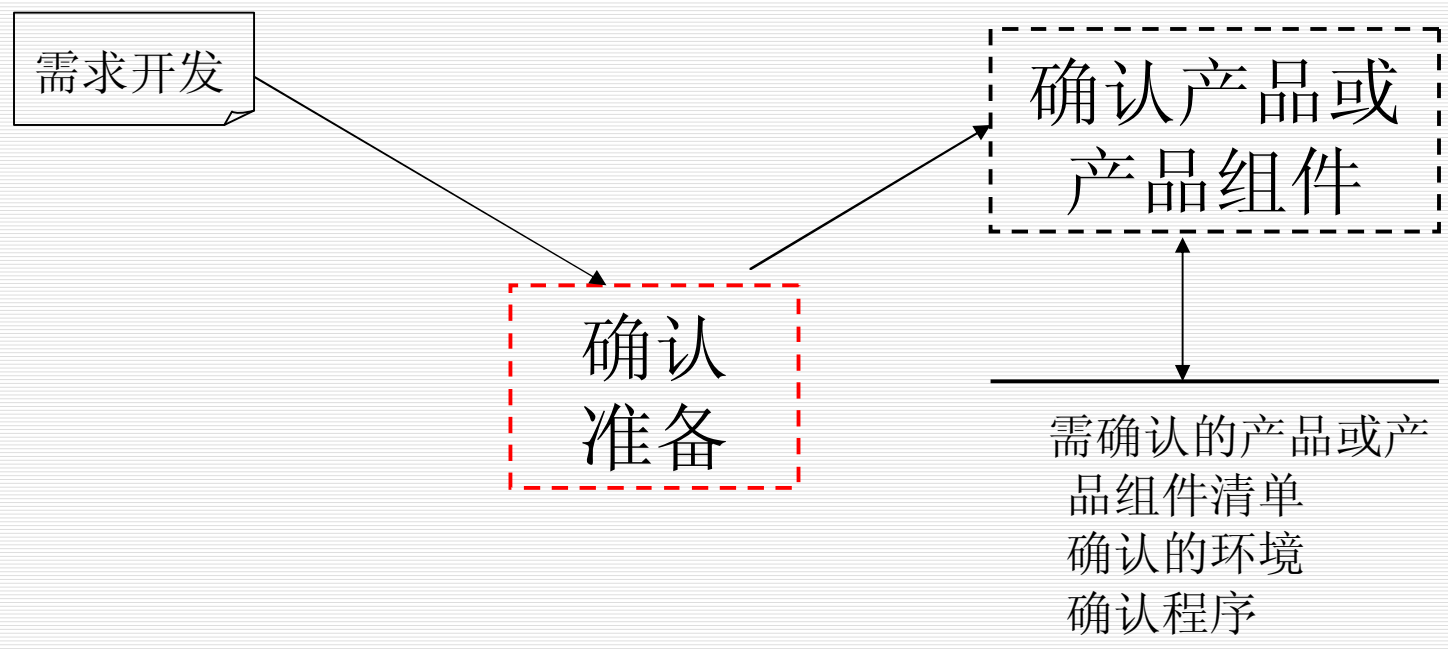
□ SG 1 确认准备

- SP 1.1 选择需确认的产品
- SP 1.2 建立确认环境
- SP 1.3 建立确认程序与准则

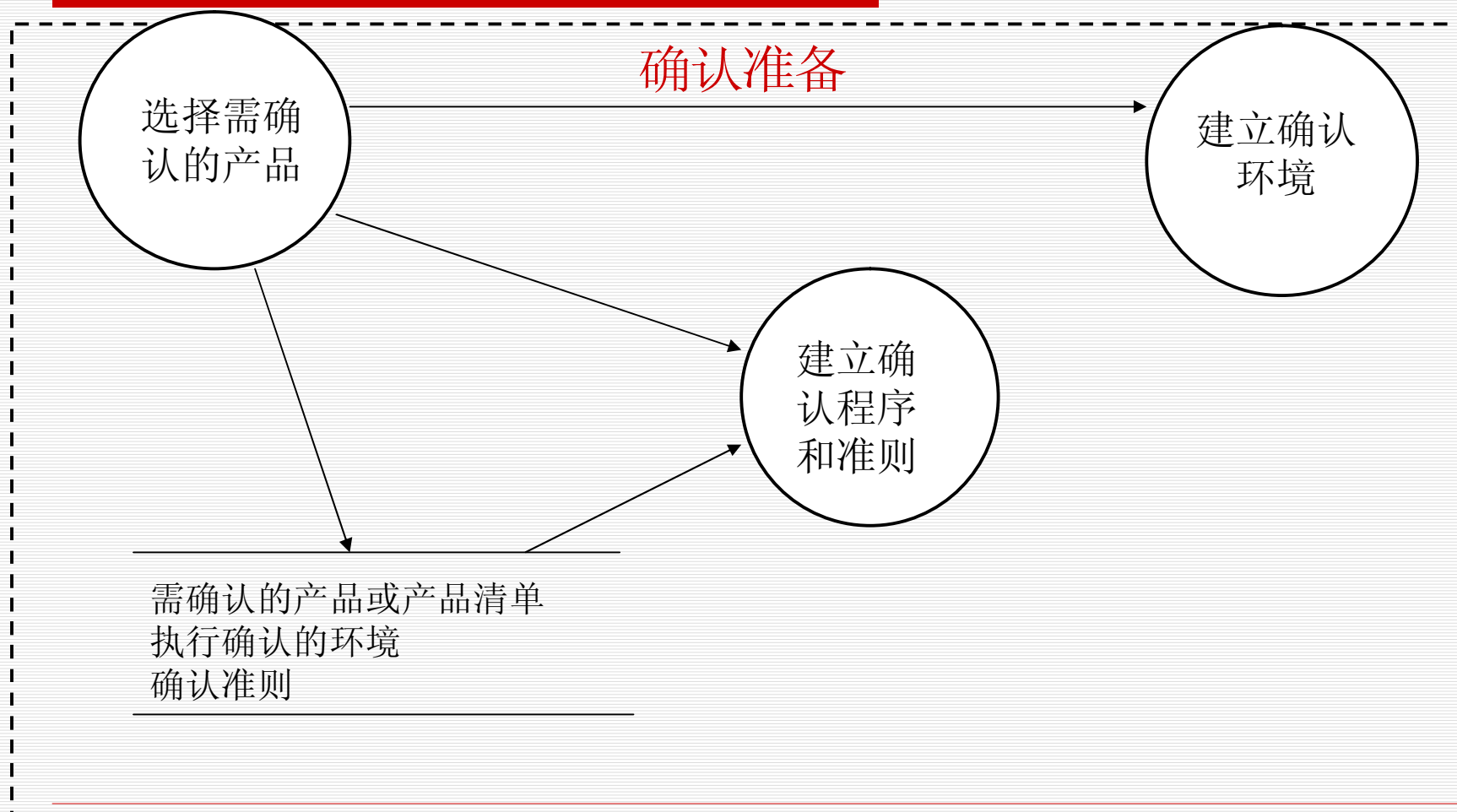
□ SG 2 确认产品或产品组件

- SP 2.1 执行确认
- SP 2.2 分析确认结果

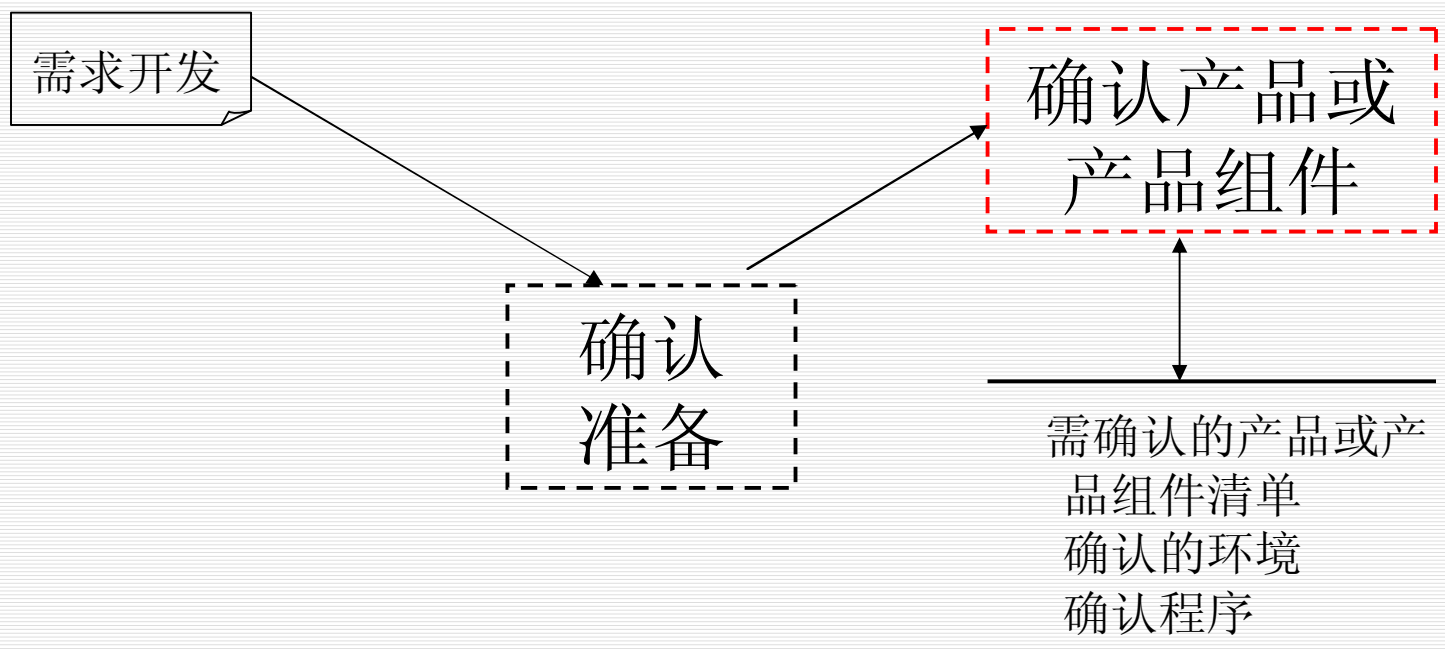
确认



确认准备



确认



确认产品或产品组件

确认产品或产品组件

执行
确认

分析确认
结果

确认报告
确认结果
确认对照表
执行程序的记录
操作展示

确认缺陷报告
确认问题
程序变更请求



确认没做好？

- 如果确认没做好，有什么现象：
 - 交付给用户的产品存在过多潜在缺陷
 - 交付给用户的产品不是用户期望的产品
 - 如果早期的确认工作被忽略，将增加产品的测试成本和返工工作量
 - 如果在交付用户前还不能发现或纠正缺陷，产品可靠性已受到损害
 - 用户产生抱怨，并不想为存在缺陷的产品买单，下次你可能就拿不到用户的合同了



VAL----总结

- ☐ 试运行
 - 试运行计划
 - 试运行过程中发现的问题记录及跟踪
 - 试运行报告
 -
- ☐ 验收测试
 - 验收计划
 - 验收过程中发现的问题记录及跟踪
 - 验收报告
 -
- ☐

谢谢！



结束语

