

软件质量保证与测试

Software Quality Assurance and Testing

第 2 章 软件测试策略

2.1 软件测试模型、阶段和生命周期



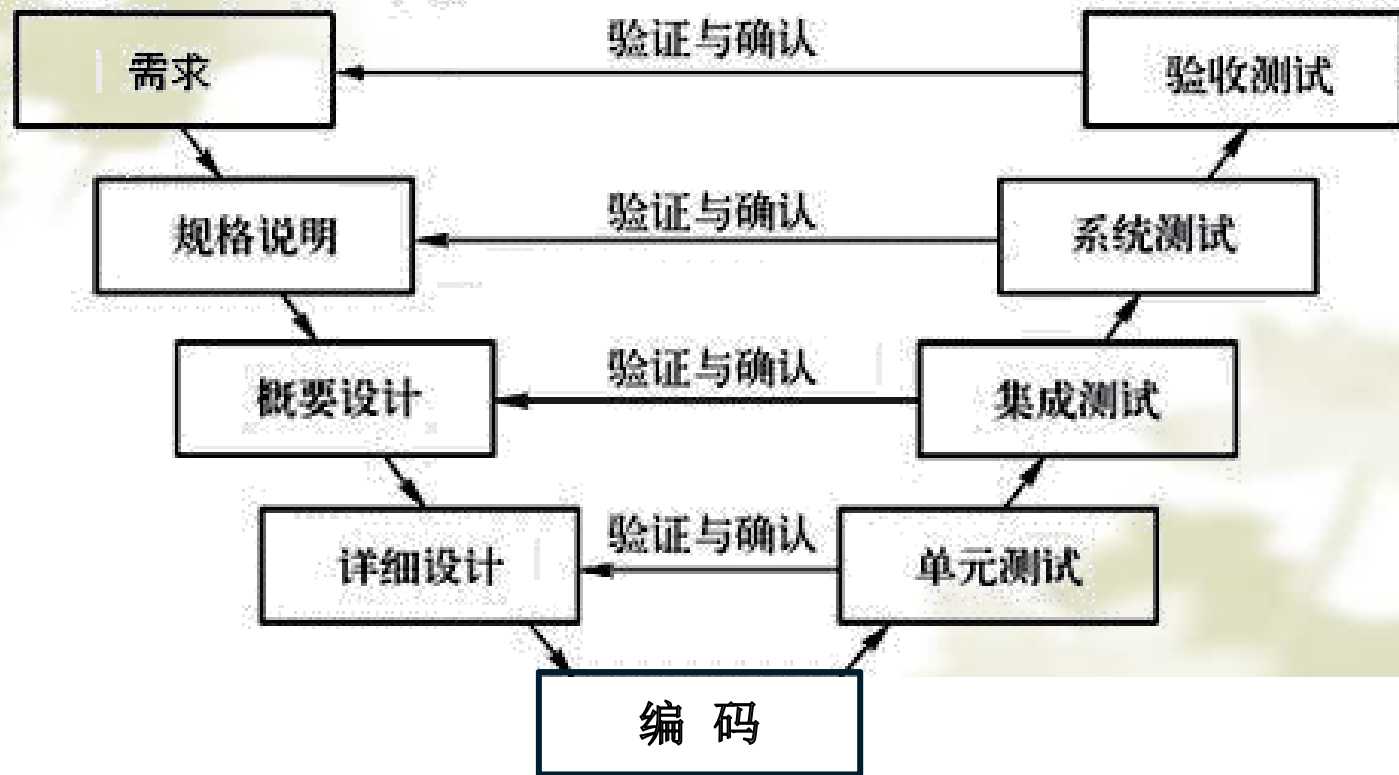
金陵科技学院

软件测试模型



我们应当 怎样来进行 软件测试？

V 模型



V 模型

表达了软件测试活动与软件分析设计活动的对应关系：

测试活动	软件分析设计活动
单元测试	详细设计
集成测试	概要设计
系统测试	软件规格说明
验收测试	软件需求

V 模型

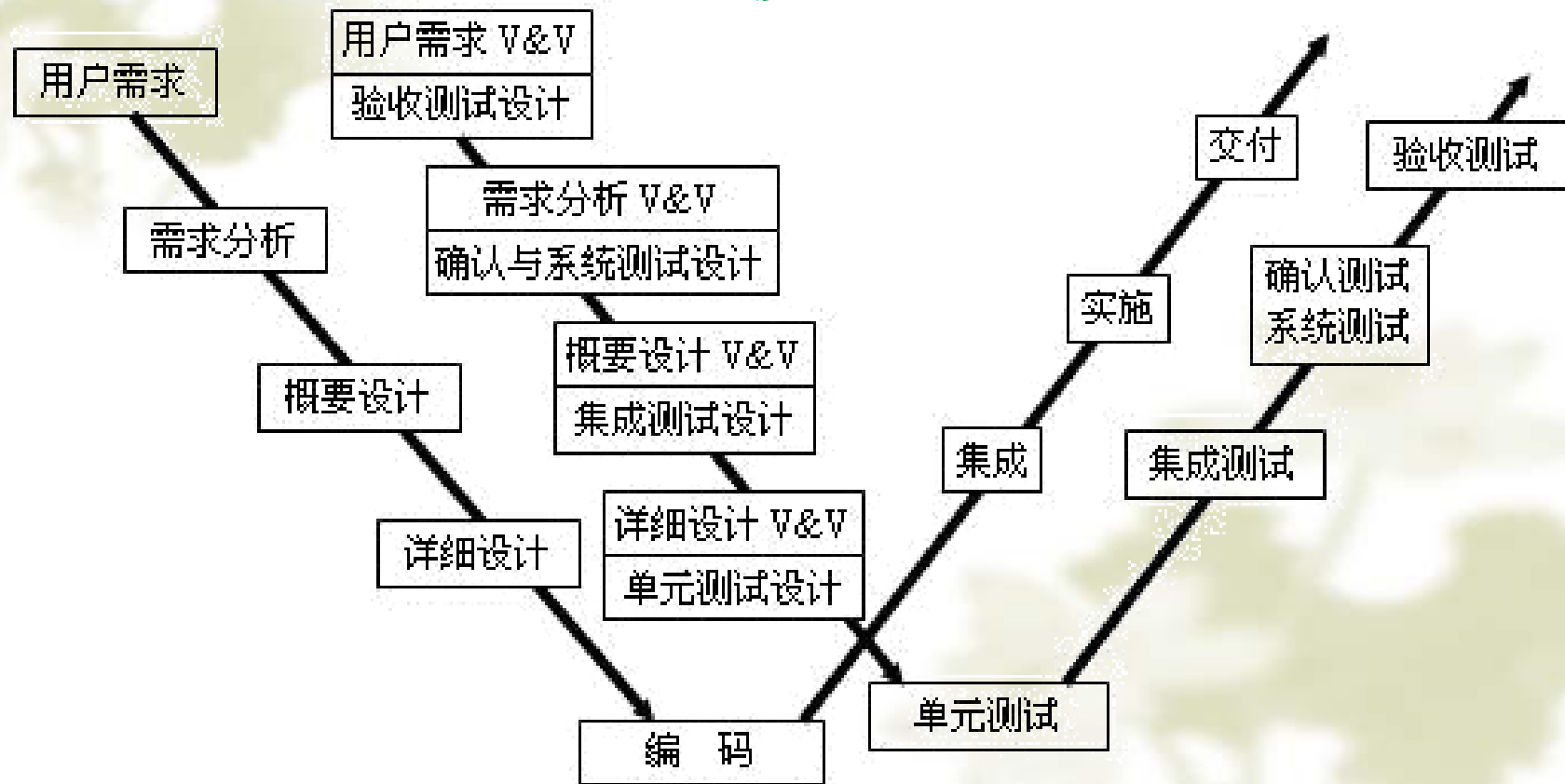
不足：

开发与测试是先后关系，先开发后测试。

忽视了对需求分析，系统设计的验证和确认，需求的满足情况一直到后期的验收测试才被验证。

如果开发阶段没有有效的质量控制措施，到软件编码完成之后，通过测试发现大量缺陷和错误，再想提高软件质量，则成本会非常高，有时甚至已经不可能。

W 模型



W 模型

相对于V模型，W模型增加了软件开发各阶段中同步进行的验证和确认活动。

W模型由两个V字型模型组成，分别代表软件质量验证、确认、测试过程和软件开发过程。

$$W = V + V$$

W 模型

W模型强调：

软件需求分析、软件设计等同样需要质量控制，应当及时进行验证和确认。

软件需求、软件设计阶段需要为后续的软件测试工作做准备、测试与开发是同步进行的。

验证、确认和测试等软件质量控制活动伴随着整个软件开发周期。

W 模型

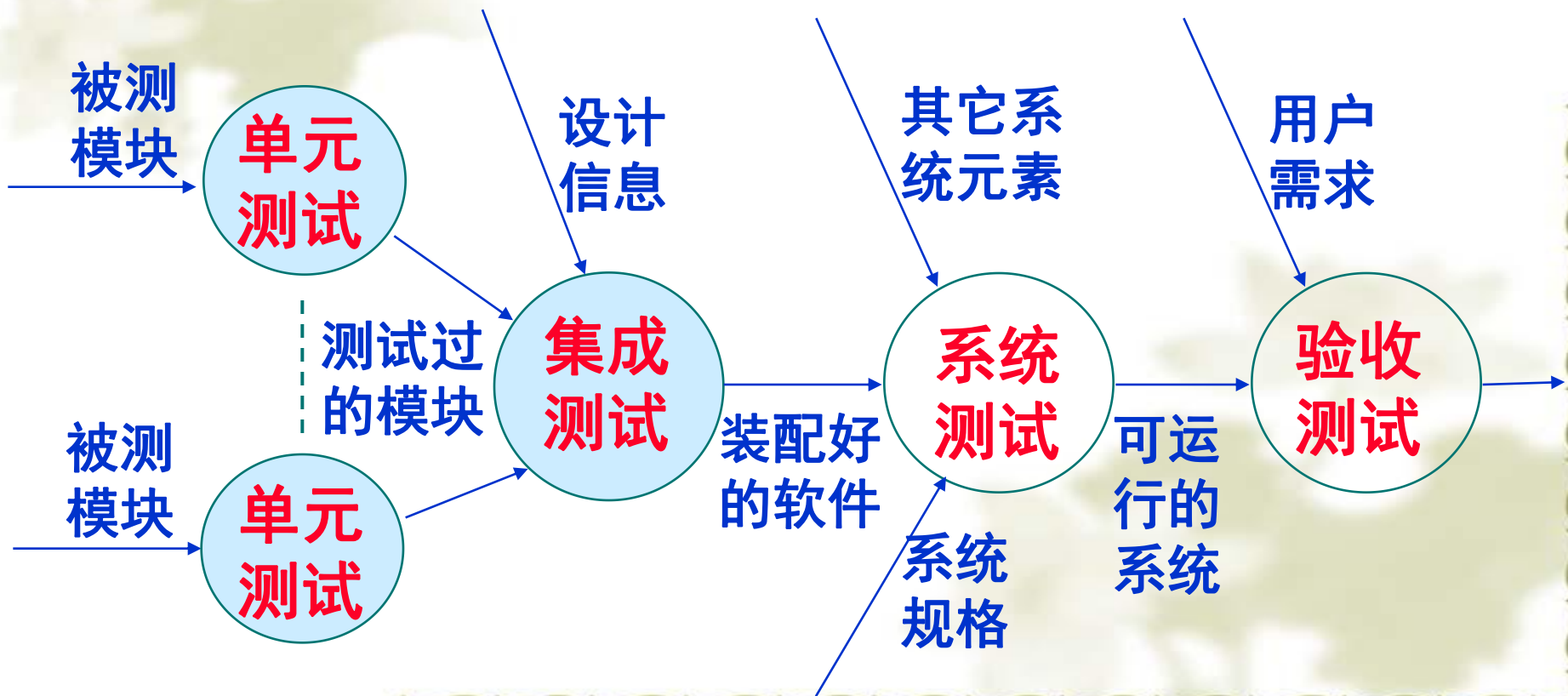
优点：

有利于尽早、全面的发现问题。例如，需求分析完成后，质量保证与测试人员就应该参与到对需求分析文档的验证和确认活动中，并尽早的发现问题。

有利于降低软件开发的总成本。因为越早发现问题，解决问题的成本就会越小。

有利于提前做好测试准备和测试设计。例如在需求分析阶段就可以及早进行验收测试设计，这将显著减少测试工作所产生的时延，加快项目进度。

软件测试的阶段



软件测试的阶段

单元测试：

是针对每个程序单元程序代码的测试，
以确保每个程序模块能正常工作为目标。

单元的粒度具体划分按不同的单位与不同的软件有不同，比如有具体到模块的测试，也有具体到类，函数的测试等。

“打印”模块

ClassA

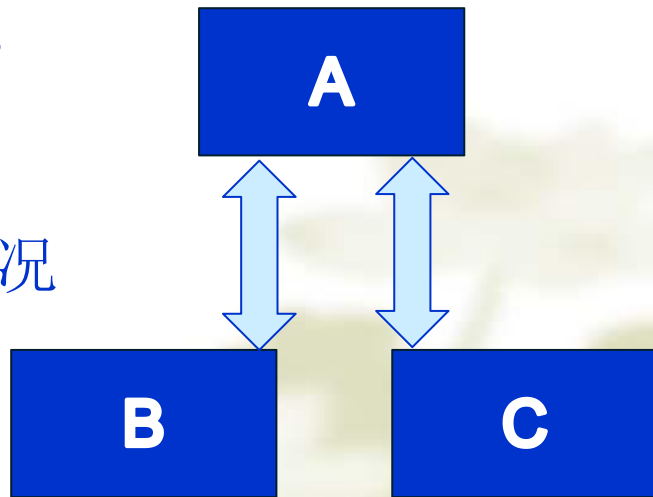
```
public int findMin( )  
{  
    .....
```

软件测试的阶段

集成测试：

对已经通过单元测试的模块，
按照设计要求进行组装和测试。

- 各模块间组合后的功能实现情况
- 模块接口连接的成功与否
- 数据传递的正确性等



软件测试的阶段

系统测试：

把软件系统搭建起来，检验软件产品能否与系统的其他部分（如硬件、操作系统、数据库等）协调工作，达到软件规格说明书中的功能、性能等方面要求。



软件测试的阶段

验收测试：

从用户的角度对软件产品进行检验和测试，看是否符合用户的要求。

验收测试可以分成Alpha测试和Beta测试。Alpha测试是由用户在开发环境下完成的测试，Beta测试是由用户在用户环境下完成的测试。

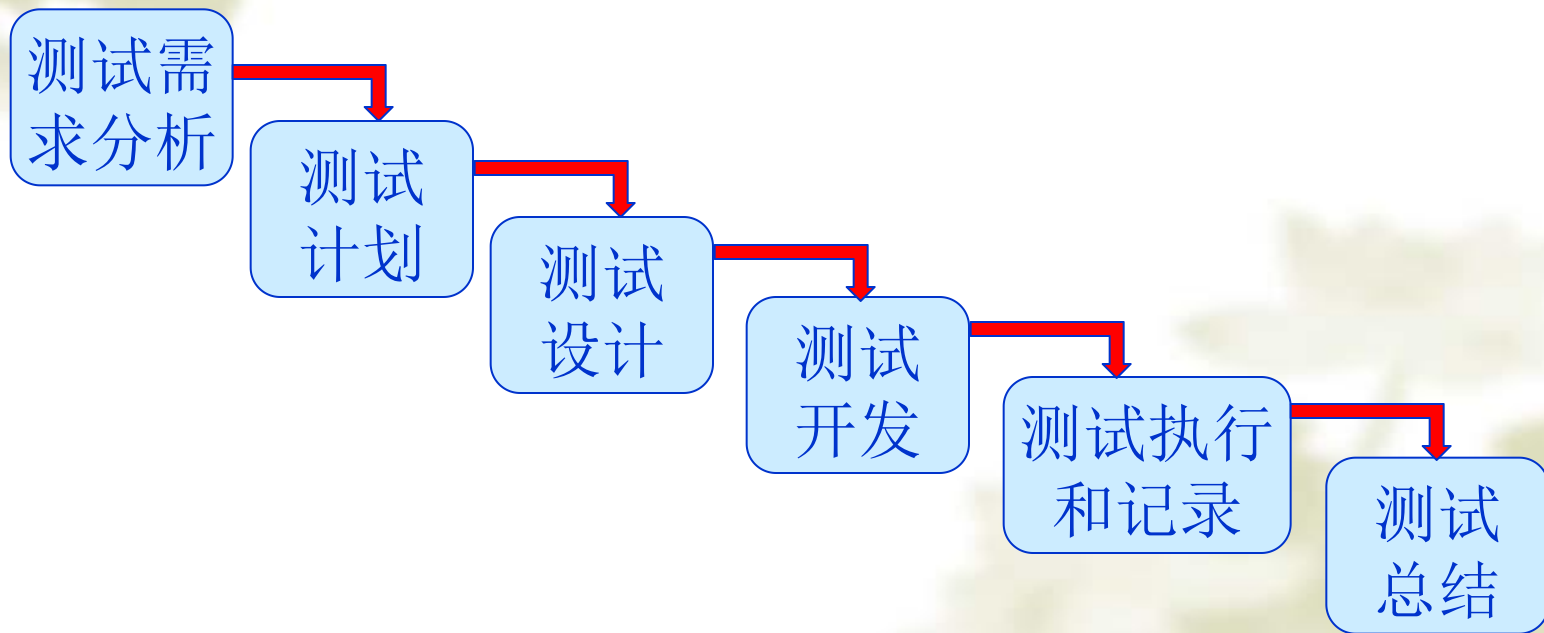


软件测试的阶段

软件测试的不同阶段，被测试对象和测试依据是不同的。

	被测试对象	测试依据
单元测试	程序模块	详细设计
集成测试	装配好的多个软件模块	概要设计
系统测试	软件系统 (包括软件及其运行环境)	软件规格说明
验收测试	可运行的软件系统	软件需求说明 以及其他用户要求

软件测试的生命周期



软件测试的生命周期

测试需求分析：

明确需要完成的测试任务、测试内容和要达到的测试要求。

测试需求可以由软件文档获取，例如软件的规格说明书中明确了软件具有某项功能，那么就需要测试这项功能是否实现。

测试需求除了有功能测试需求之外还可以有非功能测试需求，如性能测试需求、安全性测试需求。

软件测试的生命周期

测试计划：

描述所有要完成的测试工作，包括被测试项目的背景、目标、范围、方式、资源、进度安排、测试组织，以及与测试有关的风险等方面。



软件测试的生命周期

制定软件测试计划可以从以下几方面促进测试工作的开展：

1. 使软件测试工作有据可依，按部就班，进行更顺利
2. 使软件测试工作有章可循，更易于管理
2. 促进项目参与人员彼此的沟通交流，分工合作
3. 及时发现测试工作中的问题和不足，适时调整进度，资源投入，人员安排等。

软件测试的生命周期

测试设计：

如何合理运用测试原则、方法、策略，设计测试方案和数据，尽可能降低测试成本，并尽可能多的发现软件中的缺陷和问题。

测试设计要兼顾测试的充分性和成本节约原则，综合运用多种测试方法、策略，合理设计测试数据，用尽可能少的测试数据发现尽可能多的软件缺陷和问题，减少测试工作量，提高测试效率。

软件测试的生命周期

测试开发：

主要指开发测试脚本，有时也包括自动生成测试数据等。

软件测试需要重复执行软件，以便发现软件中的问题，测试开发的重要工作就是编写得到用于自动执行测试过程的代码，一般称之为测试脚本。



有时在需要大量测试数据的情况下，也可以编写程序或者通过其他工具自动生成一些测试数据。

测试开发：测试脚本实例

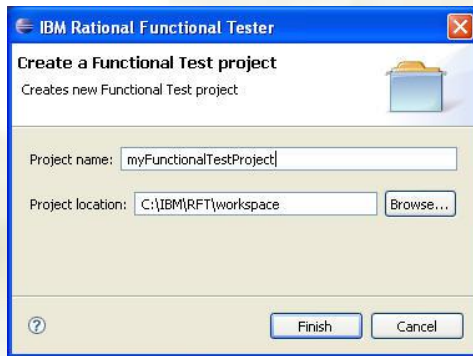
```
import resources.OrderTotalHelper;
import com.rational.test.ft.*;
import com.rational.test.ft.object.interfaces.*;
import com.rational.test.ft.script.*;
import com.rational.test.ft.value.*;
import com.rational.test.ft.vp.*;
/**
 * Description   : Functional Test Script
 * @author Administrator
 */
public class OrderTotal extends OrderTotalHelper
{
    .....
    .....
    .....
```


软件测试的生命周期

测试执行和记录:

执行测试过程，包括执行程序，输入测试数据，记录测试结果等。

目前一般都是采用自动化的方法执行测试过程。



软件测试的生命周期

测试总结:

包括统计分析测试结果，报告缺陷，评估软件质量等。

测试统计表

项目	统计数据
测试用例总数	
测试用例覆盖率	
执行测试用例数	
测试用例执行率	
已通过的测试用例数	
未通过的测试用例数	
软件缺陷密度	

缺陷报告

