



IBM 软件学院
IBM Software Institute

软件测试

IBM Rational 测试解决方案

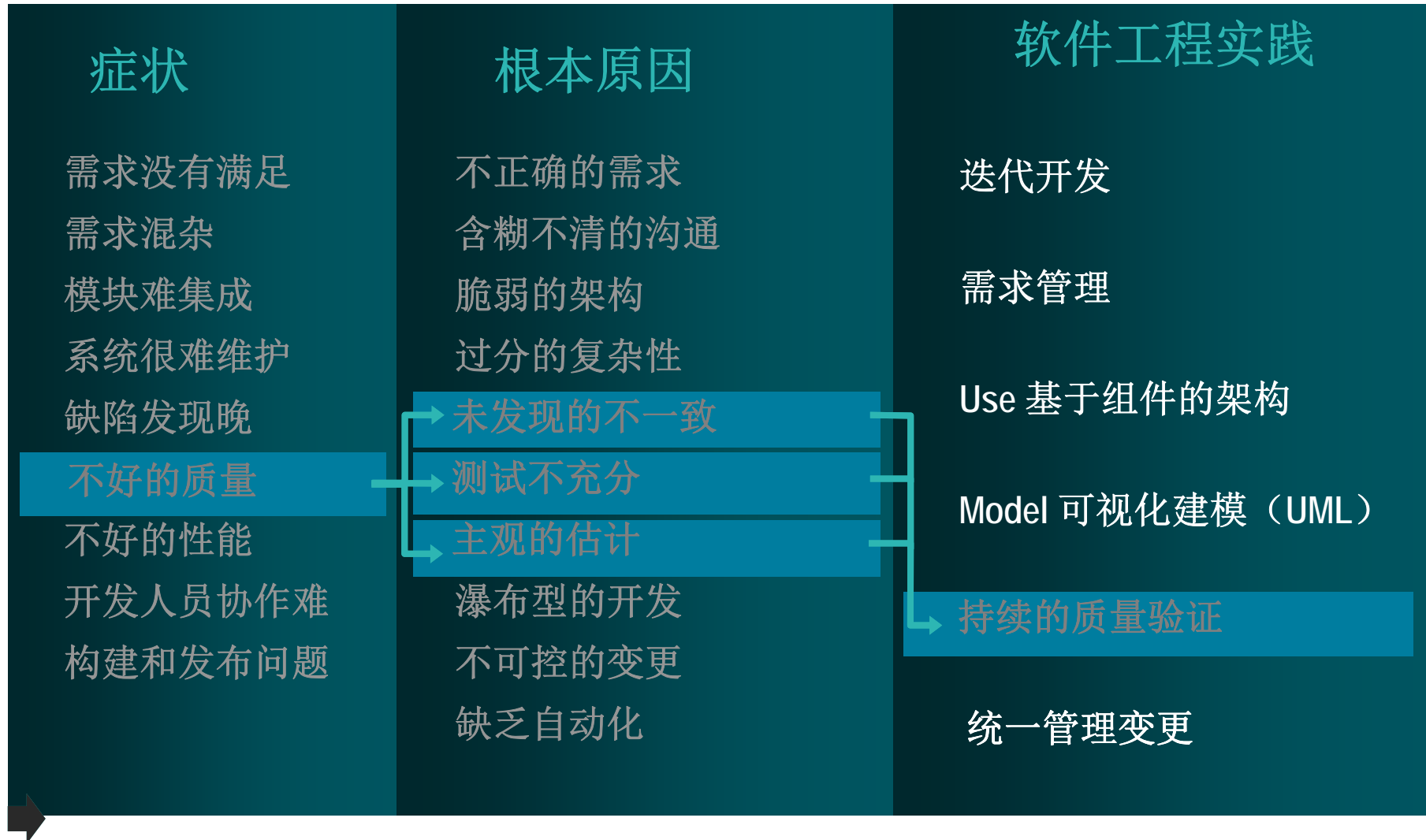
软件开发问题的症状

- 用户或者业务需求没有被满足
- 需求的混杂
- 系统模块无法集成
- 系统很难维护
- 较晚的发现系统的缺陷

软件开发问题的症状

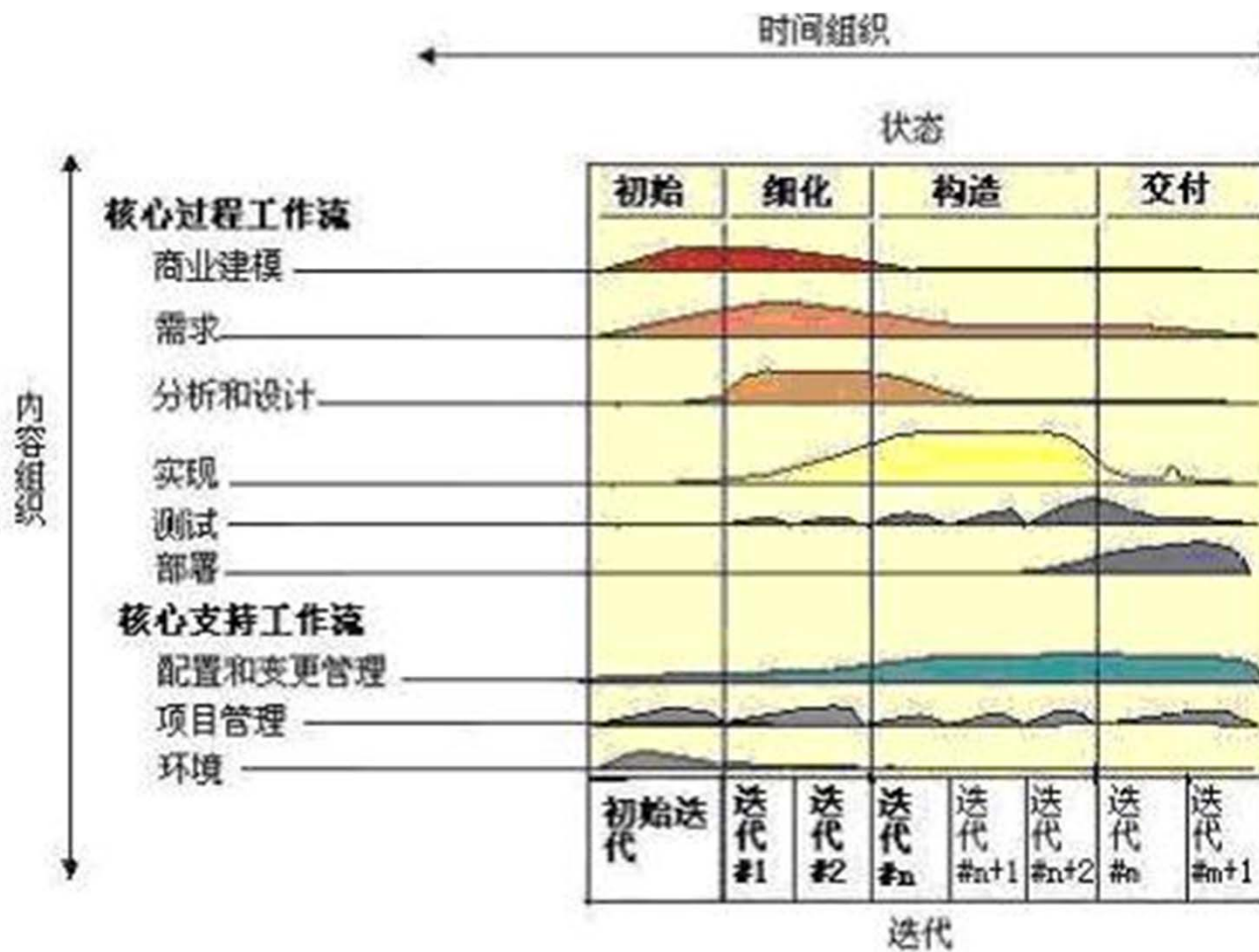
- 不好的质量或者不好的用户体验
- 不好的负载性能
- 非并行的团队工作
- 构建和发布版本的问题

从症状追溯到根本原因



RUP (Rational Unified Process) 过程

- RUP中的软件生命周期在时间上被分解为四个顺序的阶段：初始阶段(Inception)、细化阶段(Elaboration)、构造阶段(Construction)和交付阶段(Transition)
- 每个阶段结束于一个主要的里程碑(Major Milestones)；每个阶段本质上是两个里程碑之间的时间跨度
- 在每个阶段的结尾执行一次评估以确定这个阶段的目标是否已经满足。如果评估结果令人满意的话，可以允许项目进入下一个阶段。



IBM Rational 测试解决方案

- 传统软件测试过程中的问题
 - IBM Rational 软件测试最佳成功经验
 - IBM Rational 测试解决方案的整体架构
 - IBM Rational 测试解决方案

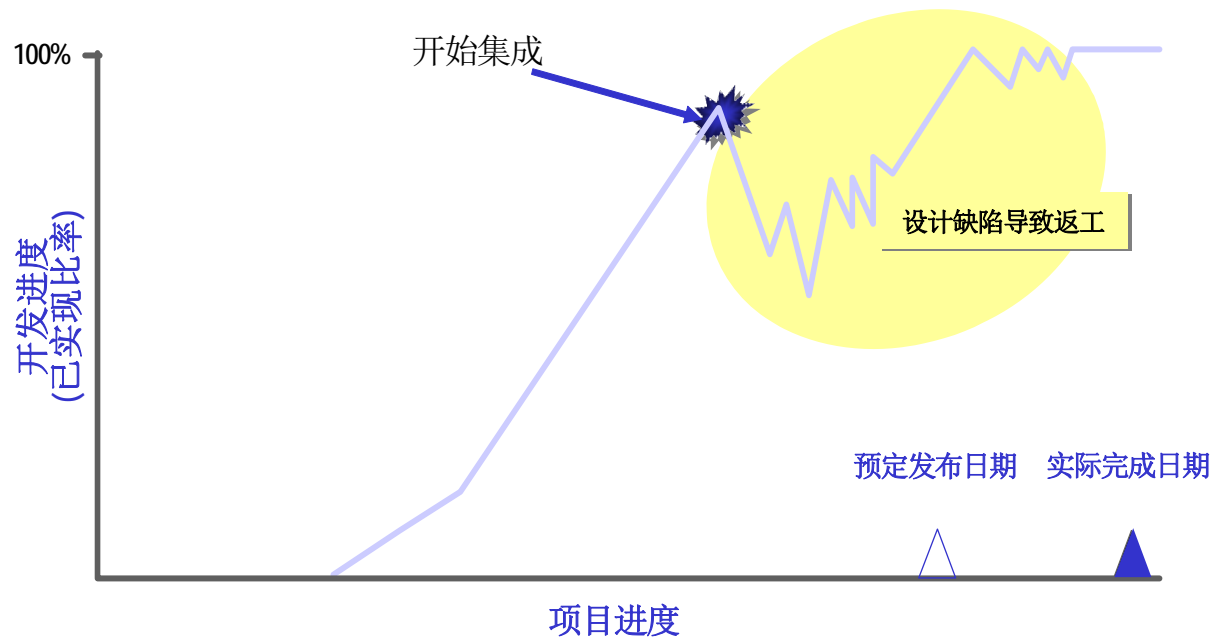
传统软件测试过程中的问题

- 问题一：项目进度难于控制，项目管理难度加大
- 问题二：对于项目风险的控制能力较弱
- 问题三：软件项目开发费用超出预算

传统软件测试过程中的问题

传统测试流程的问题

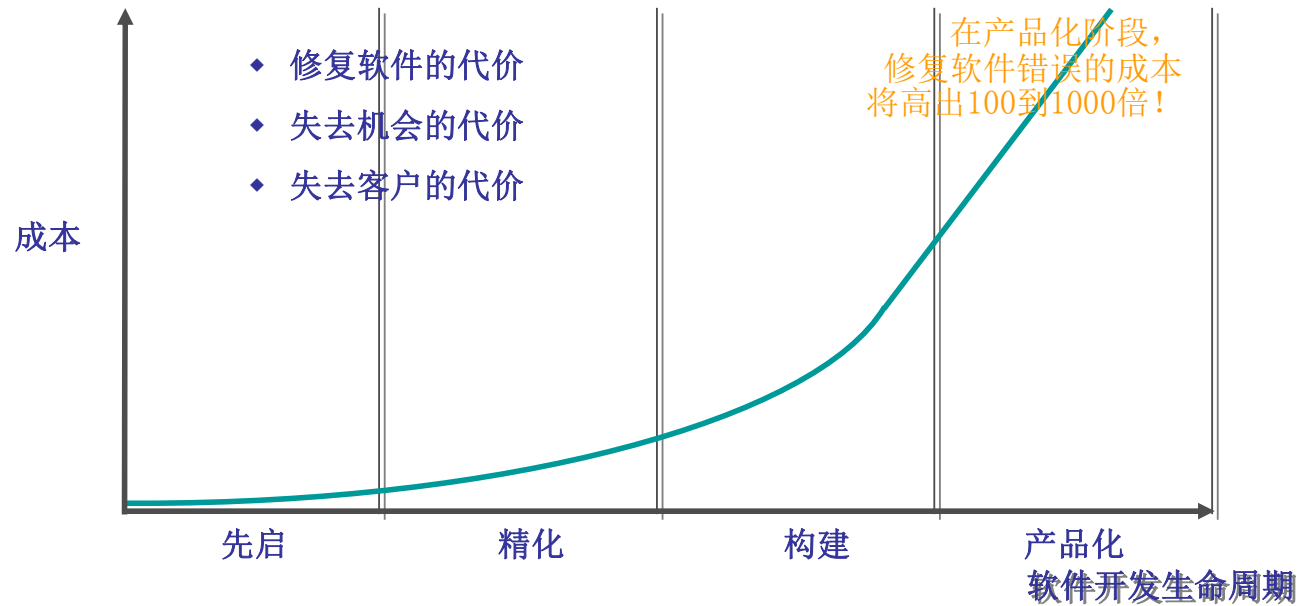
- ◆ 项目进度难于控制
- ◆ 项目风险控制能力较弱
 - ◆ 40% 的开发精力花在集成和测试上



传统软件测试过程中的问题

传统测试流程的问题

◆ 软件项目开发费用超出预算并常常延期完成



IBM Rational 测试解决方案

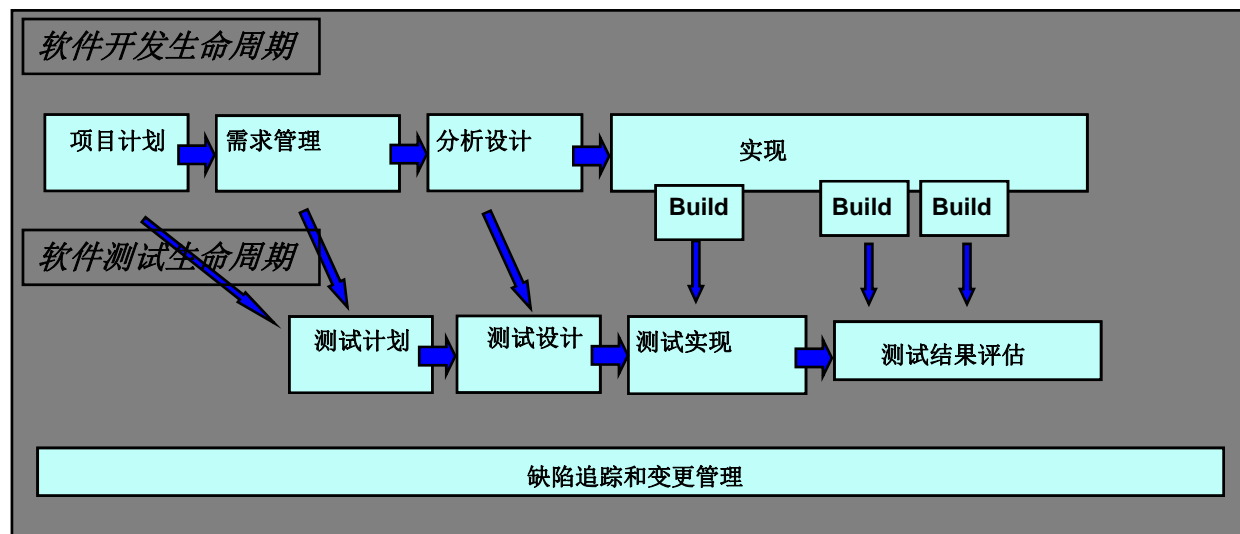
- 传统软件测试过程中的问题
- **IBM Rational 软件测试最佳成功经验**
- IBM Rational 测试解决方案的整体架构
- IBM Rational 测试解决方案

IBM Rational 软件测试最佳成功经验解决传统测试问题

- 成功经验一：尽早测试
- 成功经验二：连续测试
- 成功经验三：自动化测试

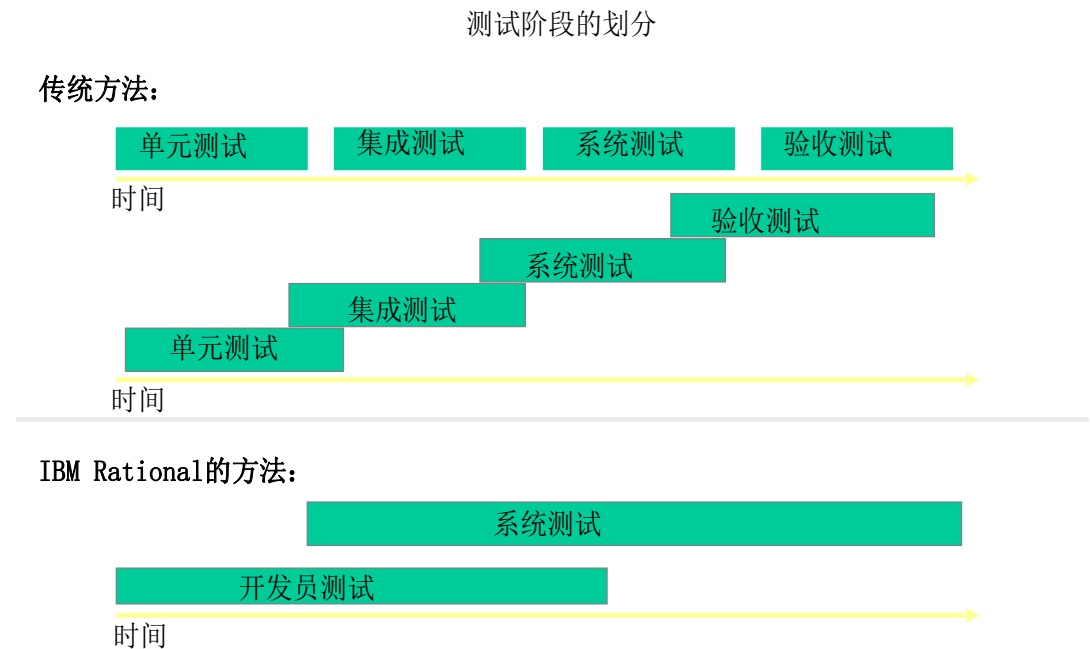
成功经验一：尽早测试

- IBM Rational 提供的尽早测试的软件工程技术：
 - 软件的整个测试生命周期是与软件的开发生命周期基本平齐的过程



成功经验一：尽早测试

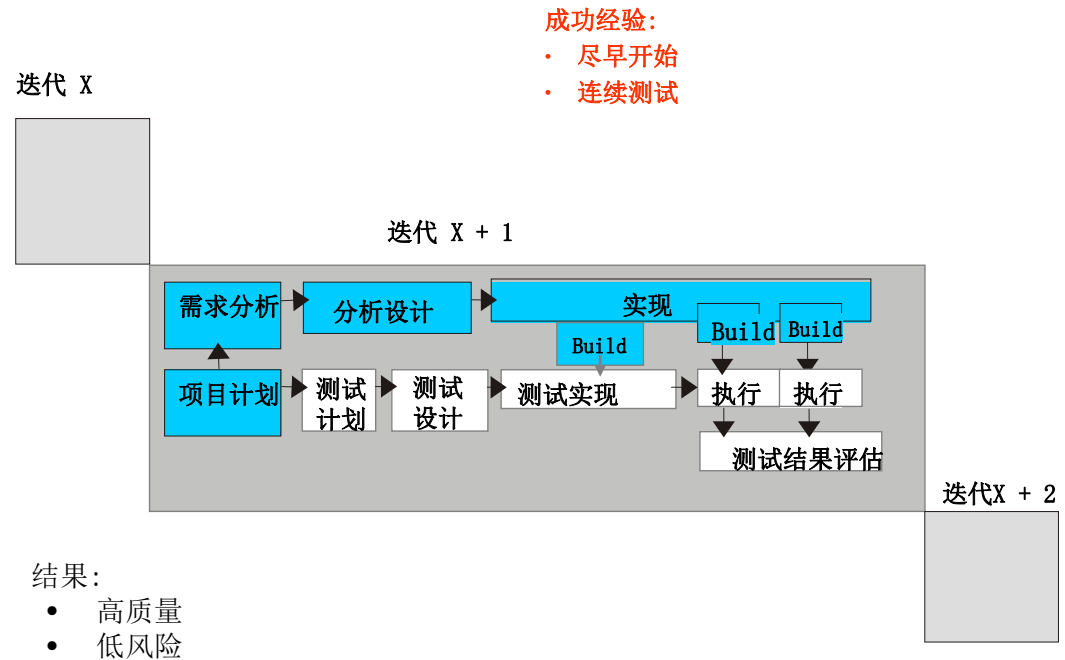
- IBM Rational提供的尽早测试的软件工程技术：
 - 通过迭代是软件开发把原来的整个软件开发生命周期分成多个迭代周期
 - 将整个软件的测试按阶段划分成开发人员测试和系统测试两个阶段



成功经验二：连续测试

- 从迭代式软件开发模式
得来
- 每一个迭代都定义明确的
阶段性评估标准
- 每一次迭代都产生一个
可运行的系统
- 在迭代式软件开发的每个
迭代周期我们都会进行软件测试活动

IBM Rational的测试成功经验：**尽早测试、连续测试**



成功经验三：自动化测试

- 要想实现尽早测试、连续测试，完善的测试流程是前提，自动化测试工具是保证。
- IBM Rational 在提供了测试成功经验之外，还为我们提供了一整套的软件测试流程和自动化测试工具，使软件测试团队能够从容不迫地完成整个测试任务。

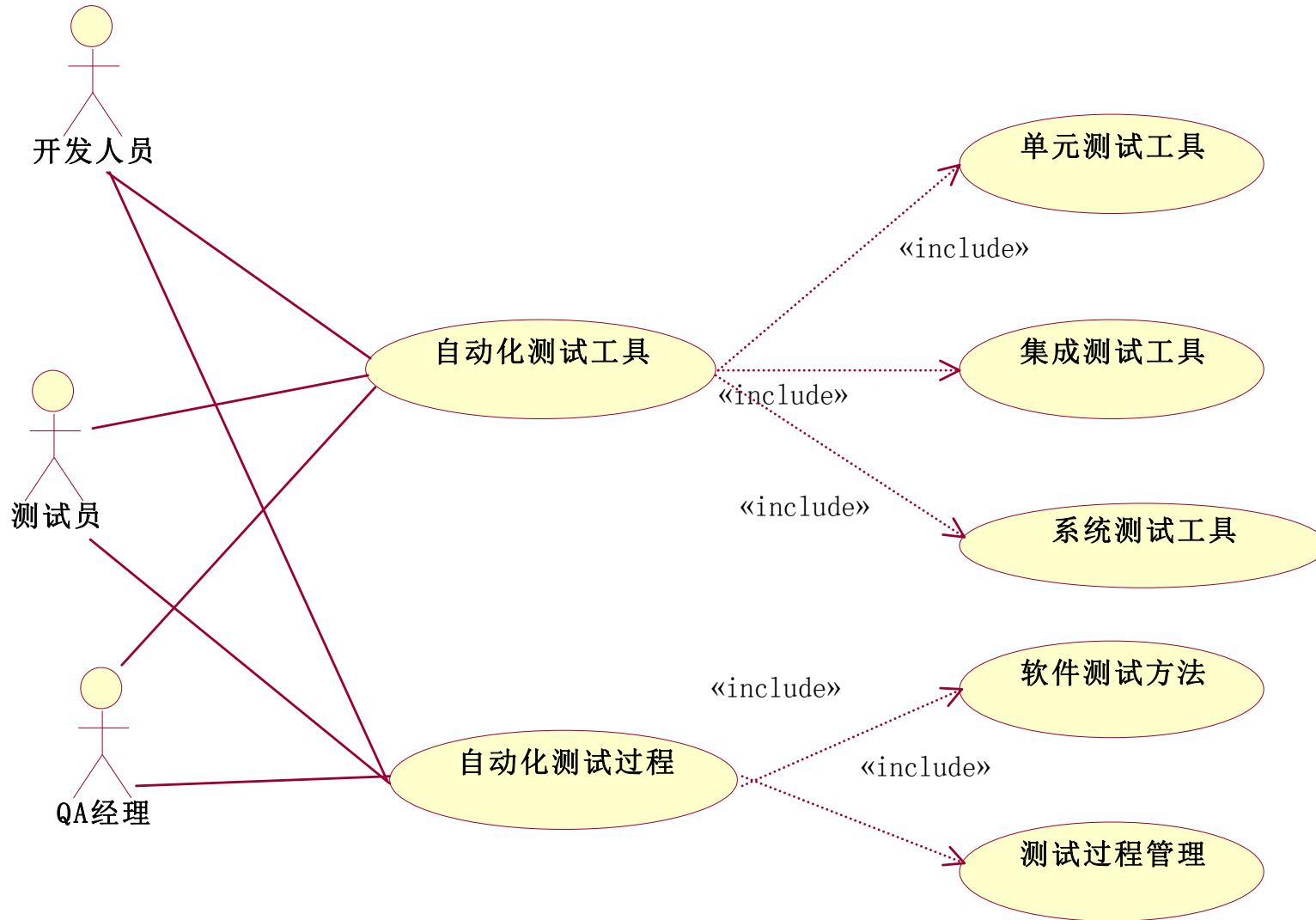
IBM Rational 测试解决方案

- 传统软件测试过程中的问题
- IBM Rational 软件测试最佳成功经验
- **IBM Rational 测试解决方案的整体架构**
- IBM Rational 测试解决方案

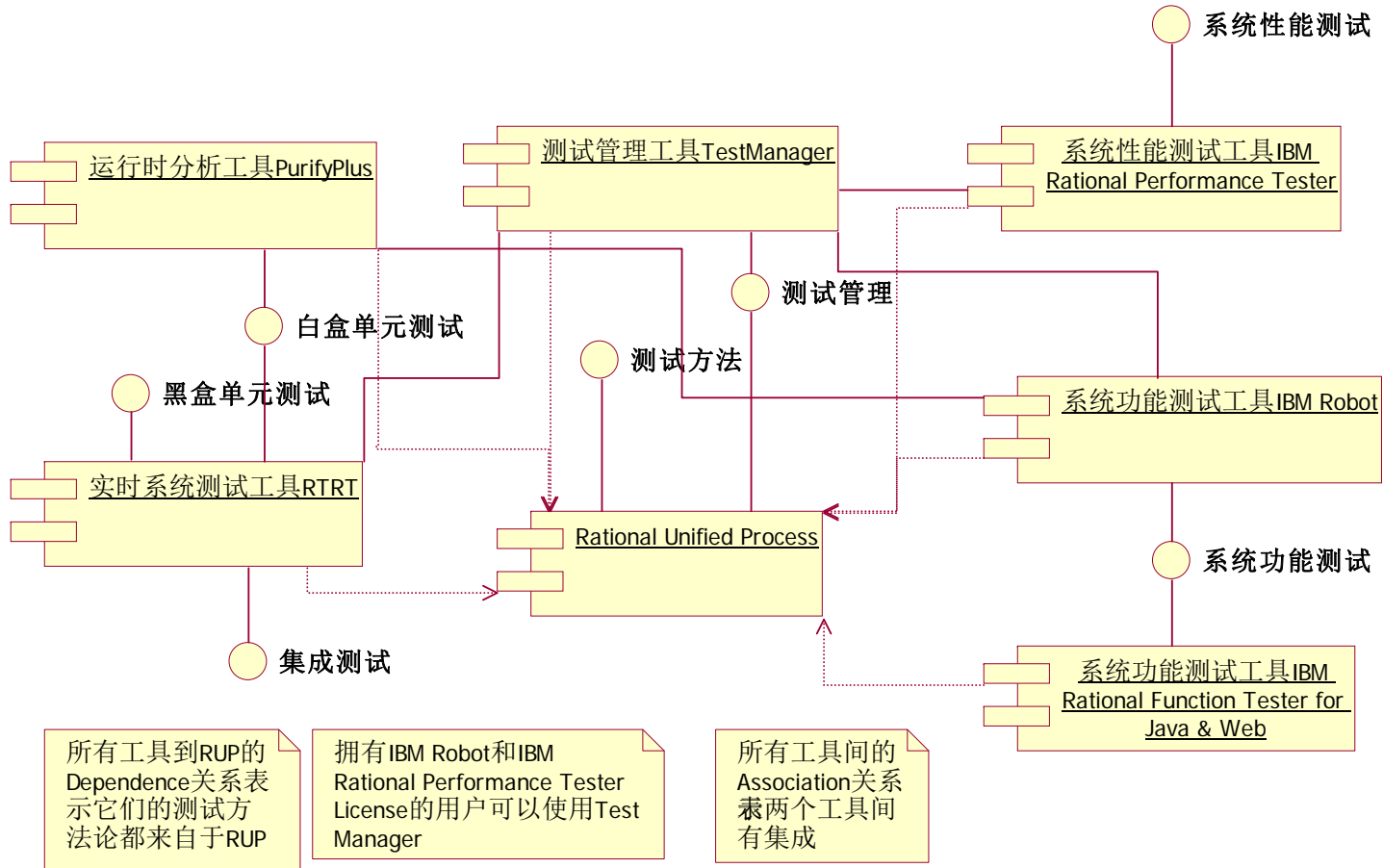
IBM Rational 测试解决方案的整体架构

- IBM Rational 测试解决方案的用例视图
- IBM Rational 测试解决方案的实现视图

IBM Rational 测试解决方案的用例视图



IBM Rational 测试解决方案的实现视图



IBM Rational 测试解决方案

- 传统软件测试过程中的问题
- IBM Rational 软件测试最佳成功经验
- IBM Rational 测试解决方案的整体架构
- **IBM Rational 测试解决方案**

IBM Rational 测试解决方案

- 单元测试快速解决方案
- 针对 Java 和 Web 应用软件的功能测试解决方案
- 针对混合 GUI 环境应用软件的功能测试解决方案
- 性能测试解决方案
- 测试管理的解决方案
- 实时系统和具有较高要求的非实时系统的软件测试解决方案

单元测试快速解决方案

- 产品 — IBM Rational PurifyPlus
 - Purify
 - Quantify
 - PureCoverage
- **IBM Rational PurifyPlus** 主要适用于要在短期内快速提升单元测试中的运行时分析能力的团队，其主要特点是：见效快、使用方便、门槛低、培训时间短，开发人员2小时内即可完全掌握该软件进行测试。

单元测试快速解决方案

- Purify 主要针对软件开发过程中难于发现的内存错误、运行时错误。
- Quantify 主要解决软件开发过程中的性能问题。
- PureCoverage 提供应用程序的测试覆盖率信息。

针对 Java 和 Web 应用软件的功能测试解决方案

- 产品
 - **IBM Rational Function Tester for Java and Web**
- 三个最重要的自动化测试的特性是：
 - 专业的自动化测试脚本创建环境
 - 测试脚本在回归测试方面的灵活性和可维护性
 - 强大的测试脚本语言

针对 Java 和 Web 应用软件的功能测试解决方案

- 专业的自动化测试脚本创建环境：
 - 统一测试和开发环境
 - 多平台支持
 - 测试工件的配置管理
 - 详细的在线帮助

针对 Java 和 Web 应用软件的功能测试解决方案

- 测试脚本在回归测试方面的灵活性和可维护性：
 - ScriptAssure
 - Object mapping
 - 设置对象表示属性
 - 创建针对动态数据的验证点
- 强大的测试脚本语言：
 - 使用标准的测试脚本语言 Java 或 VB .Net

针对混合 GUI 环境应用软件的功能测试解决方案

- 产品
 - **IBM Rational Robot**
- 可以方便的解决以下功能测试问题：
 - 回归测试
 - 利用数据池方便地解决大批量数据驱动的功能测试
 - 自动化的功能测试脚本生成

性能测试解决方案

- 产品
 - IBM Rational Performance Tester
- 捕获在HTTP/HTTPS或者基于SQL协议中客户端-服务器之间的通讯
- 可以方便的解决以下性能测试问题：
 - 准确的模拟性能测试的负载模型
 - 方便地模拟大批量的虚拟用户
 - 方便地为不同的虚拟用户提供所需的大批量的测试数据
 - 准确地提供各种性能分析报告

测试管理的解决方案

- 产品
 - IBM Rational Team Unifying Platform

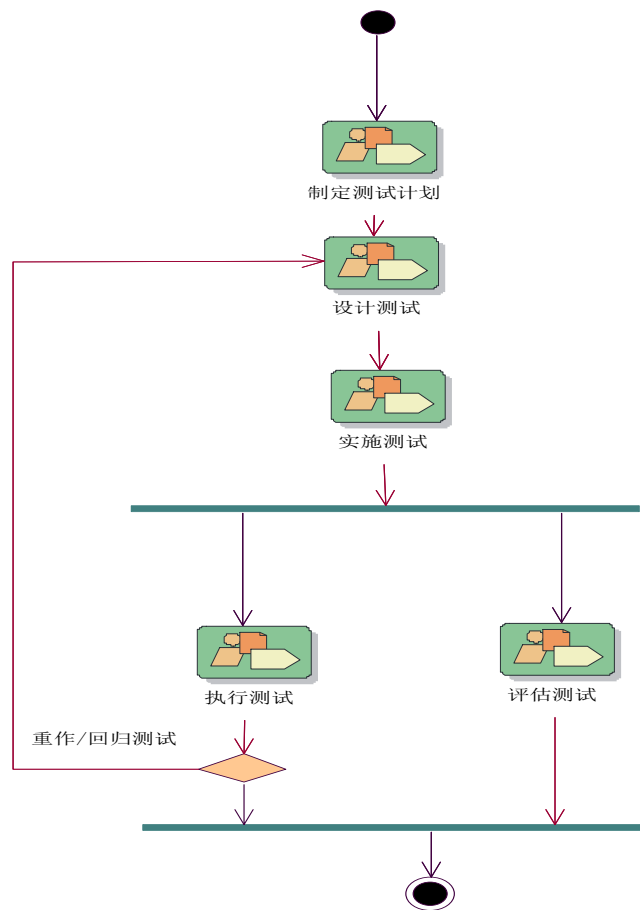
- 主要具有以下特性：
 - Rational TestManager 与 Rational 其他工具一起
 - Test Manager 作为测试管理的核心平台，统一管理从测试输入、测试计划、测试设计、测试执行和测试结果分析等整个测试过程
 - Test Manager 作为测试管理的核心平台，统一组织各种 Test Suite, Test Case, Test Script，方便地进行回归测试
 - Rational TestManager 还具有开放的可扩展API，使测试人员可以为自己专有的输入类型编写适配程序
 - 使测试人员与团队的其他成员统一

实时系统的软件测试解决方案

- 产品
 - Rational Test Realtime

- 主要具有以下特性：
 - 自动生成测试脚本模板和测试程序（包括驱动模块和桩模块）
 - 通过代码自动插针进行代码覆盖率、内存泄漏以及性能瓶颈进行分析，并和测试用例建立关联
 - 把测试结果和观察结果和被测代码关联
 - 模型观察和代码覆盖相集成
 - 与 ClearCase、ClearQuest 和 RUP 集成

IBM Rational 软件测试流程框架



本章总结

- 介绍了传统软件测试过程中存在的问题
- 介绍了IBM Rational 软件测试最佳成功经验
- 介绍了IBM Rational 测试解决方案的整体架构
- 介绍了IBM Rational 测试解决方案