



爱开发 重创新 更智慧

# Innovate2011

IBM Rational 软件创新论坛

 Software. Everywhere.



# IBM敏捷质量管理战略

Brian Bryson

Technology Evangelist, IBM Canada



# 个人简介...

- Brian Bryson

- Various roles within IBM

- Consultant

- Technical Representative

- Product Management

- Group



- Prior to IBM Rational

- University Instructor

- QA Manager

- Tester

- Accountant



- bbryson@ca.ibm.com



# 内容



- 敏捷质量
- IBM敏捷质量战略
- 小结





# 内容



## ■ 敏捷质量

- 什么是敏捷?
- 敏捷质量最佳实践



## ■ IBM敏捷质量战略

## ■ 小结



“敏捷开发的方法和相关技术正被  
全球9,000,000+（64%）开发人员所使用。”



# 什么是敏捷?

An **iterative** and incremental approach to software development which is performed in a highly **collaborative** manner by self-organizing teams within an effective governance framework, with "just enough" ceremony that produces **high quality** software in a cost effective and timely manner which meets the **changing needs** of its stakeholders.



# 什么是敏捷?

iterative

collaborative

high quality

changing needs

.





# 敏捷宣言

## Manifesto for Agile Software Development

We are uncovering better ways of developing

Individuals and interactions over processes and tools  
 Working software over comprehensive documentation  
 Customer collaboration over contract negotiation  
 Responding to change over following a plan

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	



# The Jazz Impact – Rational 自身的 改进效果

度量指标	目标	2006 评估结果	2009 评估结果
按时交付率	65%	47%	98%
缺陷修复时间	3 Months	9+ Months	3.5 months
增强需求发布率	15%	1%	21%
客户满意指数	91%	88%	96%
交付时间	12 Months	18+ Months	12.5 Months

*Note: Goals are either internal IBM statistics or industry benchmarks.*



## Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.  
Through this work we have come to value:

**Individuals and interactions** over processes and tools  
**Working software** over comprehensive documentation  
**Customer collaboration** over contract negotiation  
**Responding to change** over following a plan

**Individuals and interactions over processes and tools**

Kent Beck  
 Mike Beedle  
 Arie van Bennekum  
 Alistair Cockburn  
 Ward Cunningham  
 Martin Fowler

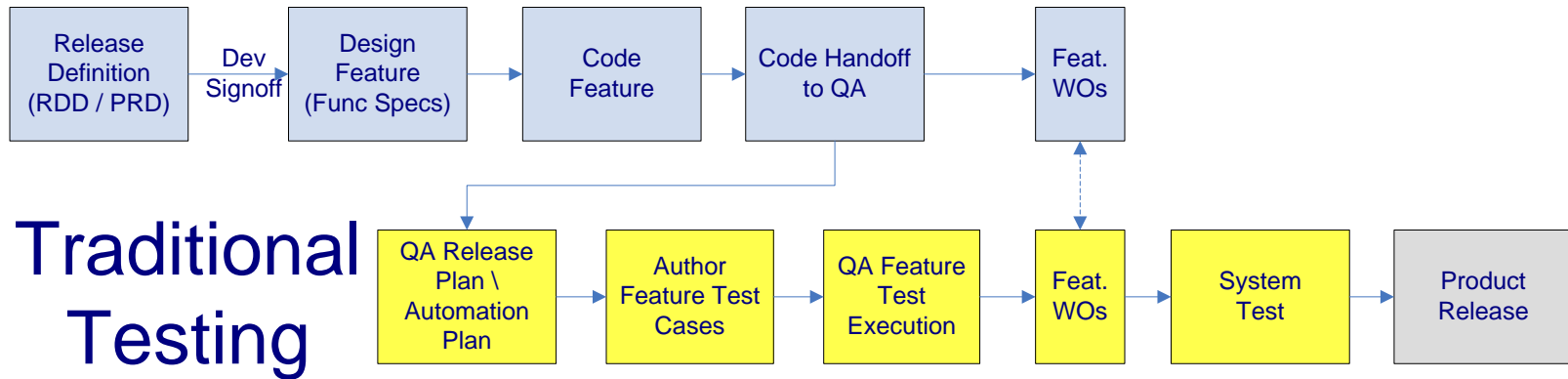
James Grenning  
 Jim Highsmith  
 Andrew Hunt  
 Ron Jeffries  
 Jon Kern  
 Brian Marick

Robert C. Martin  
 Steve Mellor  
 Ken Schwaber  
 Jeff Sutherland  
 Dave Thomas



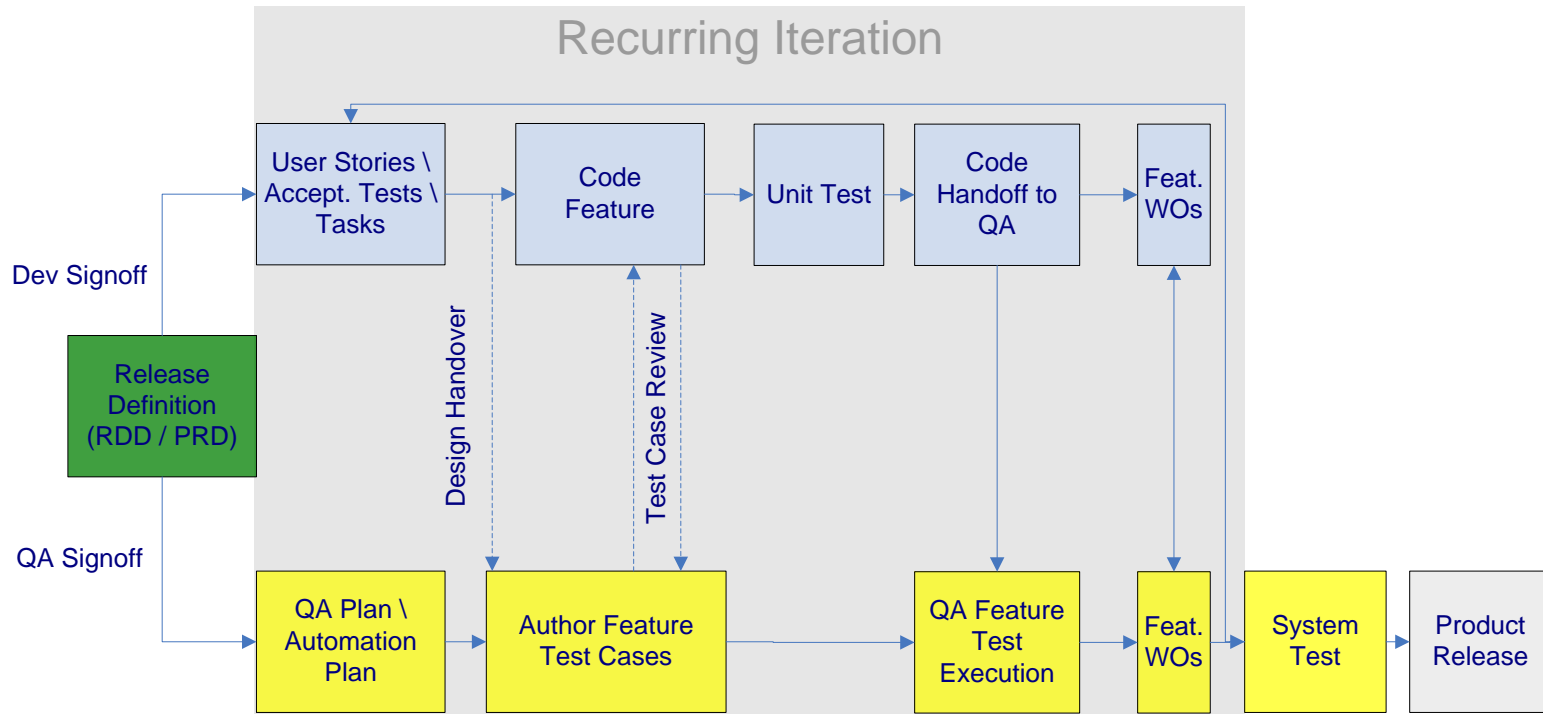


# 传统质量的解决之道



- 线性的, 顺序的
- QA 计划发生较晚
- 测试计划没有协作

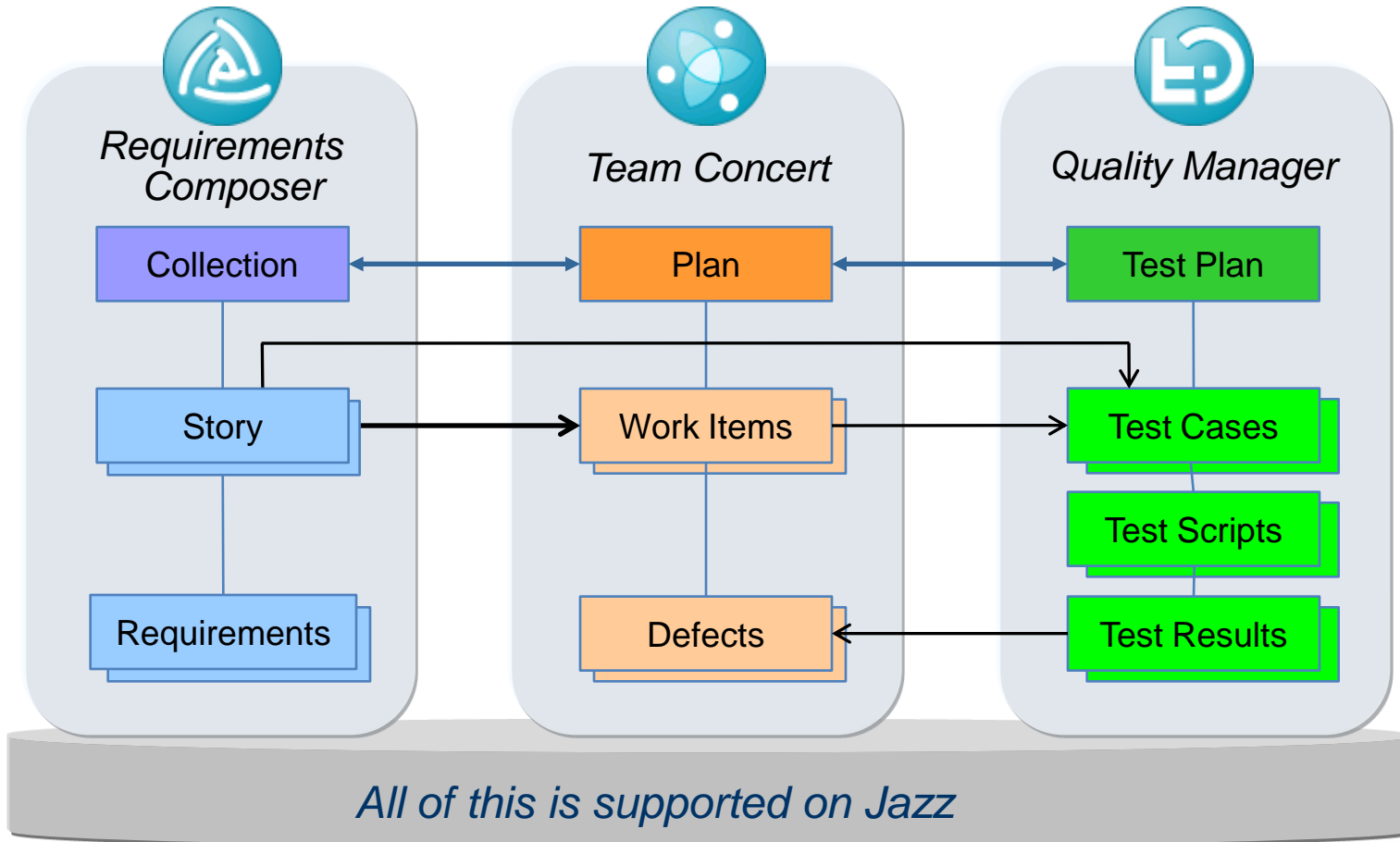
# 敏捷质量的解决之道



- 持续的测试
- 持续的反馈
- 持续的进展



# 个人、工件和交互



## Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.  
Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

**Working software over comprehensive documentation**

Kent Beck  
Mike Beedle  
Arie van Bennekum  
Alistair Cockburn  
Ward Cunningham  
Martin Fowler

James Grenning  
Jim Highsmith  
Andrew Hunt  
Ron Jeffries  
Jon Kern  
Brian Marick

Robert C. Martin  
Steve Mellor  
Ken Schwaber  
Jeff Sutherland  
Dave Thomas



# 敏捷测试计划

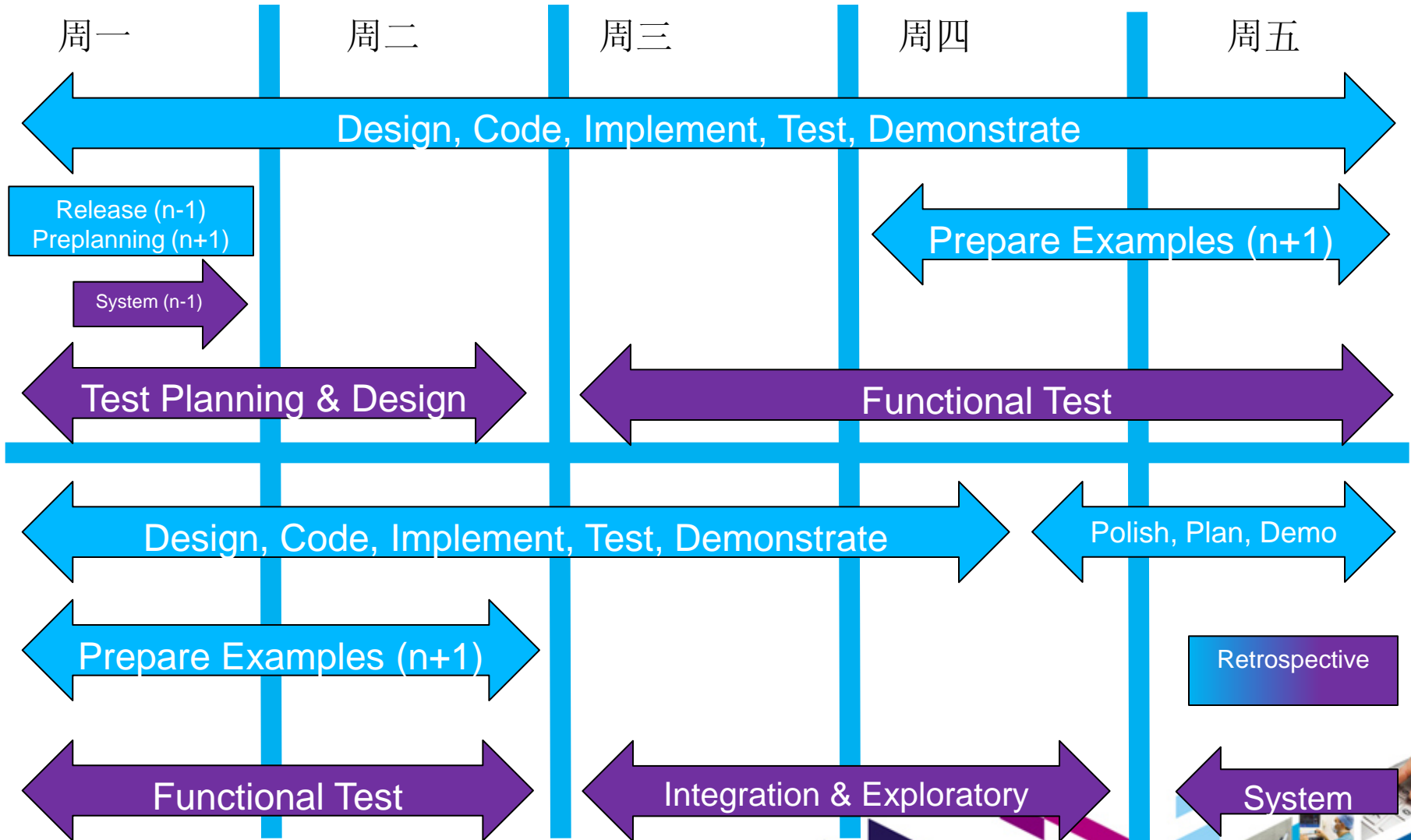
- 发布测试计划 (主测试计划)
  - 沟通测试策略
  - 质量目标
  - 资源估算
- 迭代/Sprint测试计划
  - ▶ 每个Sprint需要验证的需求
  - ▶ 测试进度
  - ▶ 测试环境和测试用例
  - ▶ 整体状态报告
  - ▶ 缺陷、问题和任务管理

*Keep it light!*

The screenshot displays two overlapping windows from the IBM Rational software interface. The top window is titled "Master Test Plan - Watson Project" and shows a "Table Of Contents" on the left with sections like Summary, Test Objectives, Requirement Collection Links, Risk Assessment, Test Schedules, Test Environments, Test Team, Quality Objectives, Test Cases, Attachments, and Child Test Plans. The main content area shows the "Summary" section with fields for Product (Discussion Forums), Release (3.1), and Test Phase (User Acceptance Test). The bottom window is titled "Child Test Plan" and shows a similar "Table Of Contents" and "Summary" section, but with the Test Phase set to "Integration Test". Both windows include a "Manage Sections" tab and a "Show All Sections" checkbox.



# 以一个2周的迭代为例



Adapted from: Gojko Adzic - gojko@gojko.net



# Agenda

## Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.  
Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

## Customer collaboration over contract negotiation


Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	



## 自动化验收测试过程

- 手工测试时产生验收测试, 利用关键字和执行变量加速自动化

### Manual Steps ?

Step	Description	Expected Results
1	<b>JKE Login</b> Log into JKE Banking with <b>username</b> and <b>password</b> username password	Accounts Overview screen appears 
2	Go to user's home page	home page displays
3	<b>Create Account</b> Select create new account. Enter current username, <b>username</b> . Enter address information. Record account number Set <b>account</b>	Validate account created successfully.
4	View account details page.	validate new account with id <b>account</b> is listed.
5	JKE Logout	

Set within a keyword script and referenced in the main script.

#### Keyword View ?

##### Clipboard ?

Type Filter Text

---

##### Steps

Add one or more steps to this view by dragging them from the Manual Test editor.

#### Execution Variables ?

Type Filter Text

Name    Value  
 username    jbrown  
 account  
 password    password123

## Manifesto for Agile Software Development

We are uncovering better ways of developing software by doing it and helping others do it.  
Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

## Responding to change over following a plan

Kent Beck	James Grenning	Robert C. Martin
Mike Beedle	Jim Highsmith	Steve Mellor
Arie van Bennekum	Andrew Hunt	Ken Schwaber
Alistair Cockburn	Ron Jeffries	Jeff Sutherland
Ward Cunningham	Jon Kern	Dave Thomas
Martin Fowler	Brian Marick	



# 保持进度的更新

基于个人和团队的任务管理

**挑战:** 在跨团队的情况下分配和协调测试计划的责任和事件

**解决方案:** 可视化承诺, 减少返工, 跟踪任务和监控和事件

测试目标, 测试用例  
分配和意见一致

需求和应用安全

测试迭代

OM Event Log (17 new)

- [9] Provide the Review and Approvals Section for Classics CD (3) 3 hours ago
- \* Login window does not appear (38) 3 hours ago
- \* Classics Request (37) 5 hours ago
- \* Provide the Test Scripts Section for New Customer Order (36) 5 hours ago

Page 1 of 3

团队事件日志

My Tasks

Id	Summary	Artifact
Authoring 3	The artifact has been approved	Classics CD
Authoring 3	Provide the Review and Approvals Section for Classics CD	Classics CD
Authoring 2	Provide the Summary Section for Classics CD	Classics Admin
Authoring 1	Provide the Summary Section for Classics CD	Classics CD

个人工作列表

知道对他人的预期, 知道他人对你的预期



# 内容

- 敏捷质量
- IBM敏捷战略
  - 已实现的
  - 计划实现的
- 小结



# 可扩展的协作生命周期管理

实时、透明地访问项目数据、风险和进度

Open Source Solutions

Rational Workbench for Collaborative Lifecycle Management

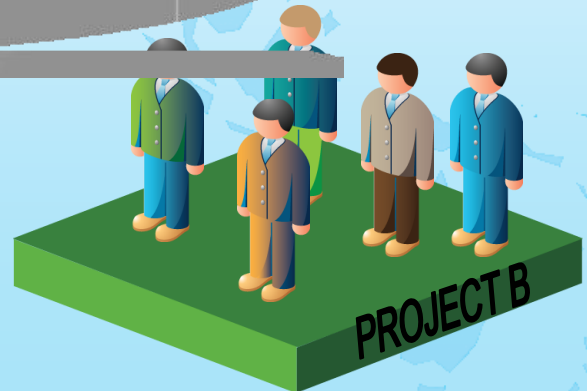
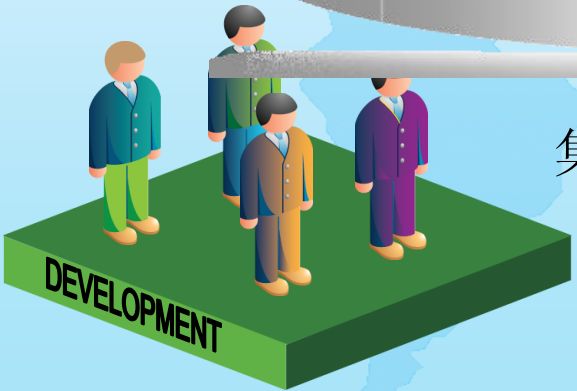
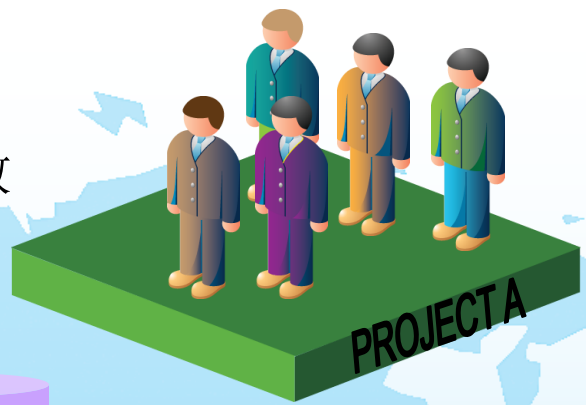
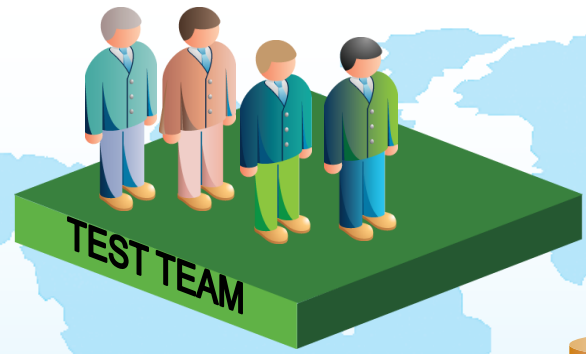
Your Existing Capabilities

Additional IBM Jazz Capabilities

IBM Partner Solutions

HTTP/REST

集中的系统管理和安全控制，灵活的部署方式







# 开放的Rational Quality Manager生态圈



## Rational software

## Tivoli software

### Automated Testing

- Rational Functional Tester
- Rational Performance Tester
- Rational Service Tester for SOA Quality
- Rational AppScan Tester Edition
- Rational Test RealTime
- Rational Rhapsody

### Builds, WorkItems and Defects

- Rational ClearQuest
- Rational Team Concert
- Rational BuildForge

### Reporting

- Rational Insight

### Provisioning

- Tivoli Provisioning Manager (TPM)
- Tivoli Service Request Manager
- TADDM

### Requirements

- Rational Requirements Composer
- Rational DOORS



Supporting bidirectional integration with the Jira change management system



Accelerating test sandbox creation through service virtualization



Managing mobile applications testing across a global handset test environment



Model-based test generation



Certify



Quick Test Professional LoadRunner



TMAP Process

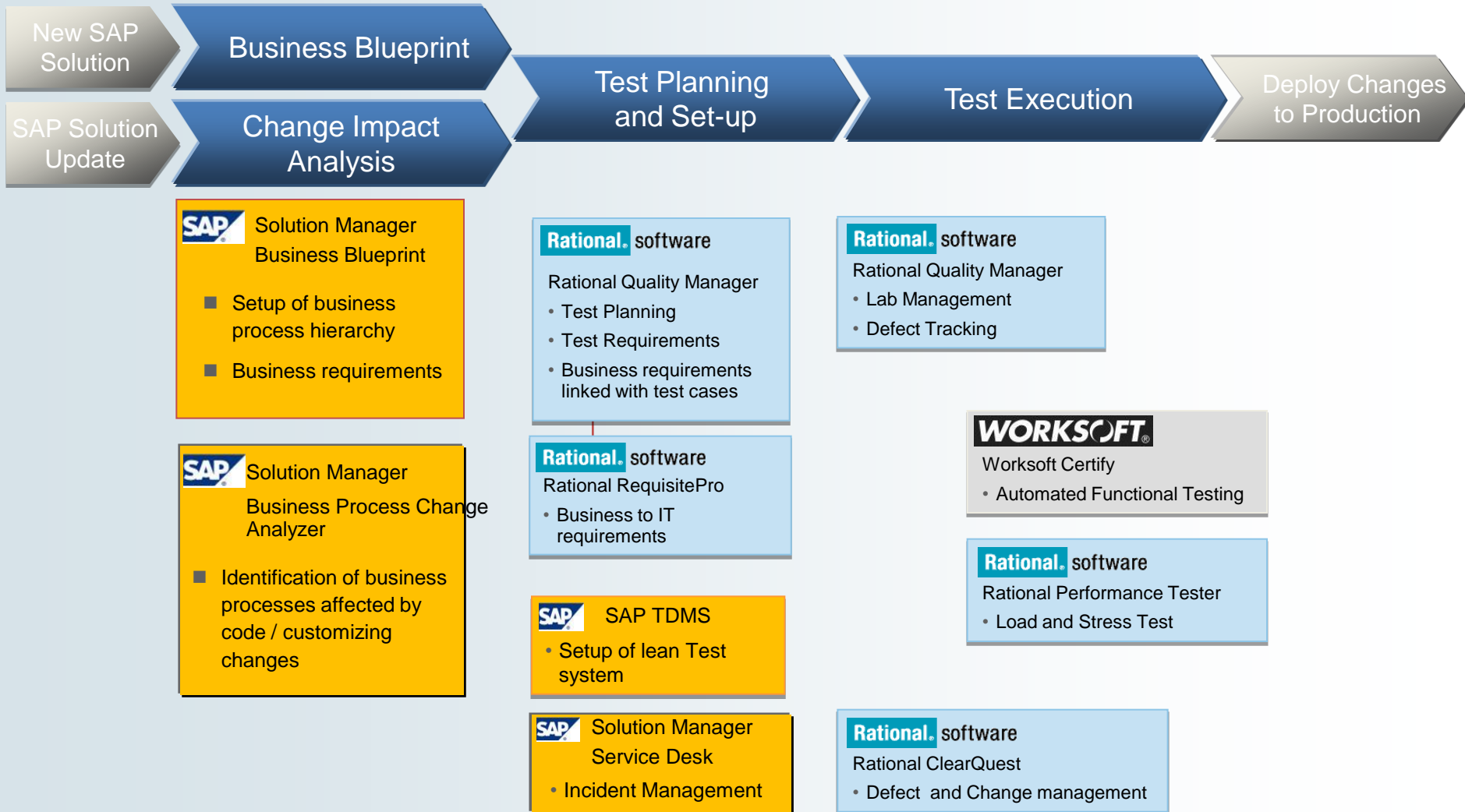


Application Performance Analysis



Telecom Test Automation

# 端到端的SAP质量管理解决方案



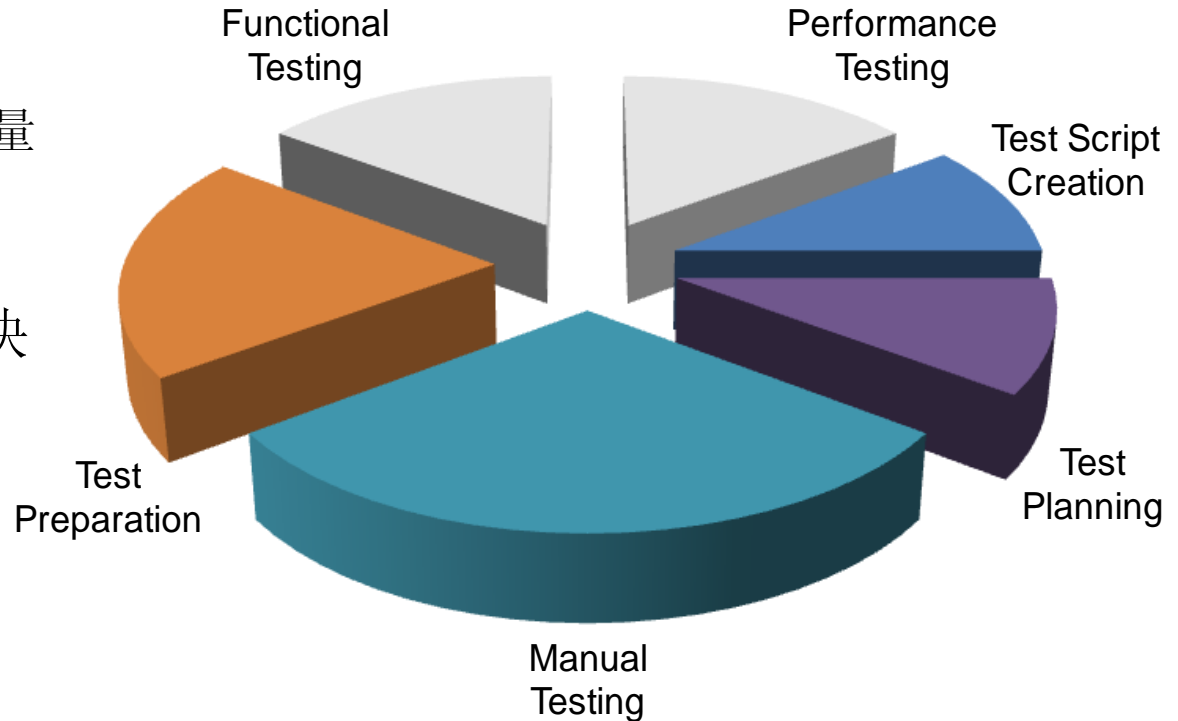
# 内容

- 敏捷质量
- IBM敏捷战略
  - 已实现的
  - 计划实现的
- 小结



# 质量保证的未公开的秘密

- 以往关注在测试自动化
  - 功能、性能、服务测试
  - 测试工作量 < 30% QA 工作量
- 没有或很少工具能够解决 50%+QA 的工作

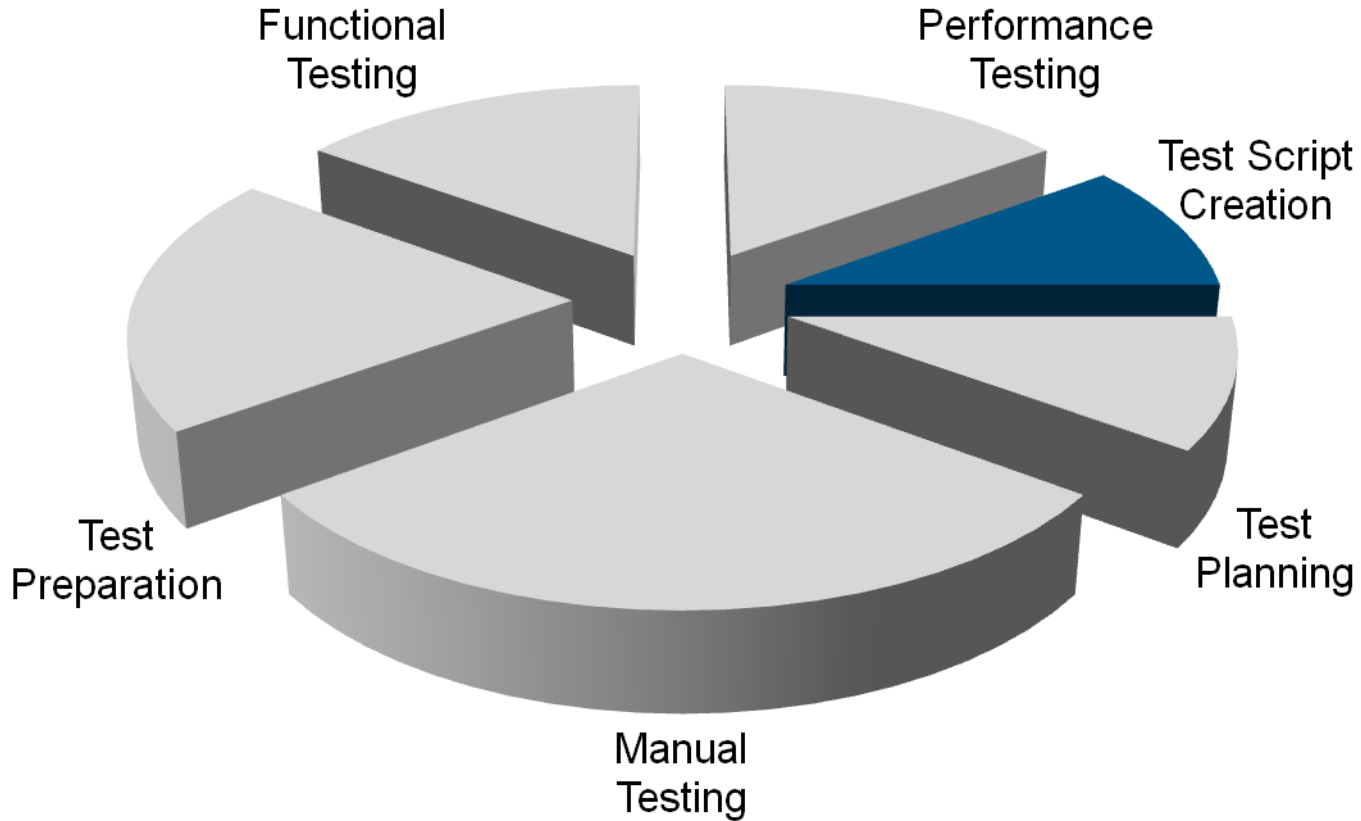


Estimated QA labor cost breakdown by activity – IBM Feb. 2011





# 测试脚本生成



# 更加智慧的测试 vs. 更多的测试

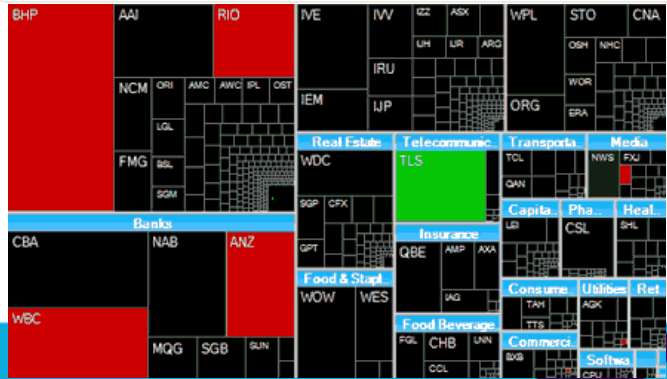
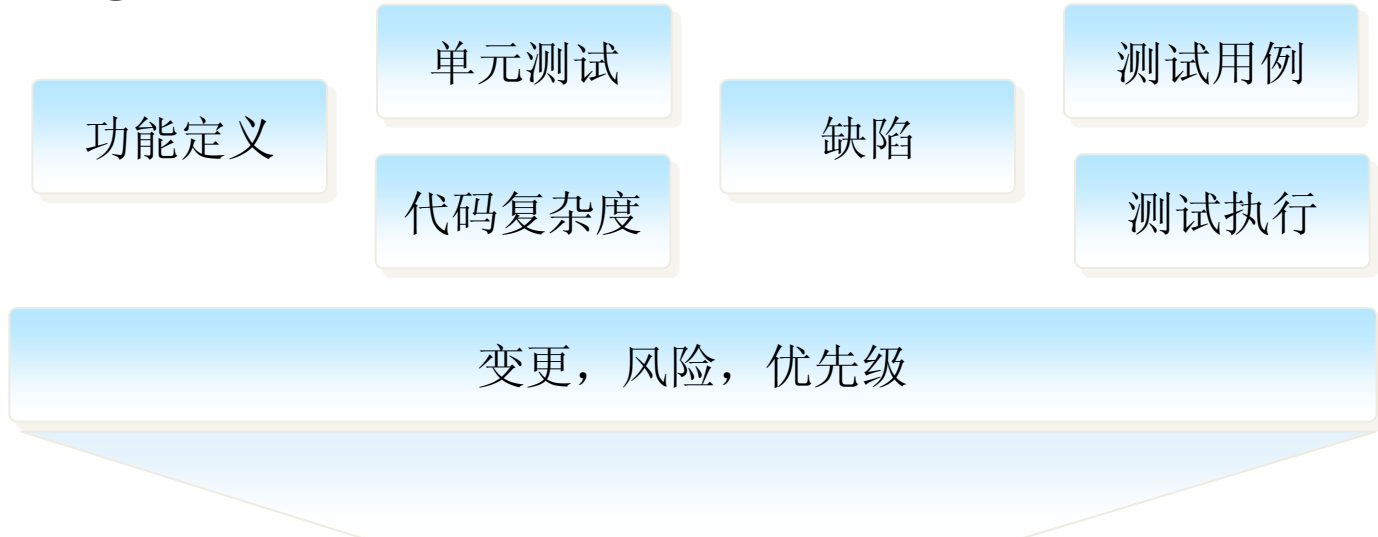
- 事实: 测试不可能面面俱到!
  - 合理的方法是: 关注正确的测试
- 研究不表明:
  - ▶ 80%的缺陷来源于20%的模块 -- Barry Boehm
  - ▶ 20% 的测试用例覆盖80%的业务价值 – Barry Boehm

如何选择正确的20%?



# 即时的测试生成指引

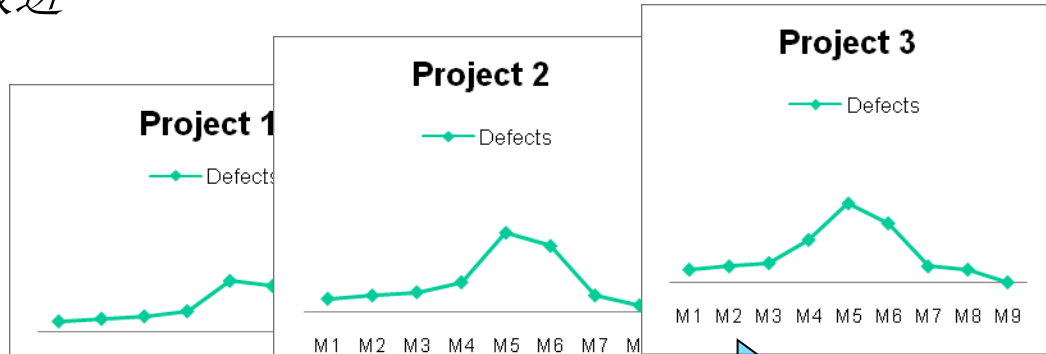
- 利用智能分析，从而进行更有针对性的测试
  - e.g. 针对有风险的功能区域进行测试



# 基于历史数据调整更好的质量流程

基于过往经验的流程改进

■ 持续的数据收集



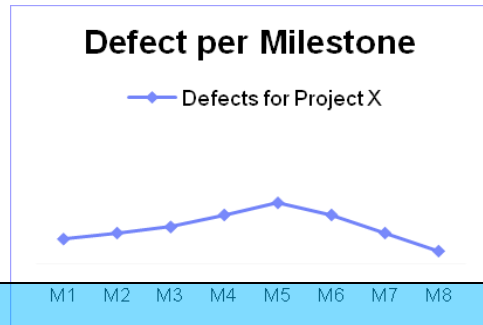
趋势显示最新的里程碑出现缺陷失效

- 对Release的影响
  - 增加时间或更多的里程碑
  - 减少内容以维持进度

■ 基于数据的智能分析



■ 实时的流程调整

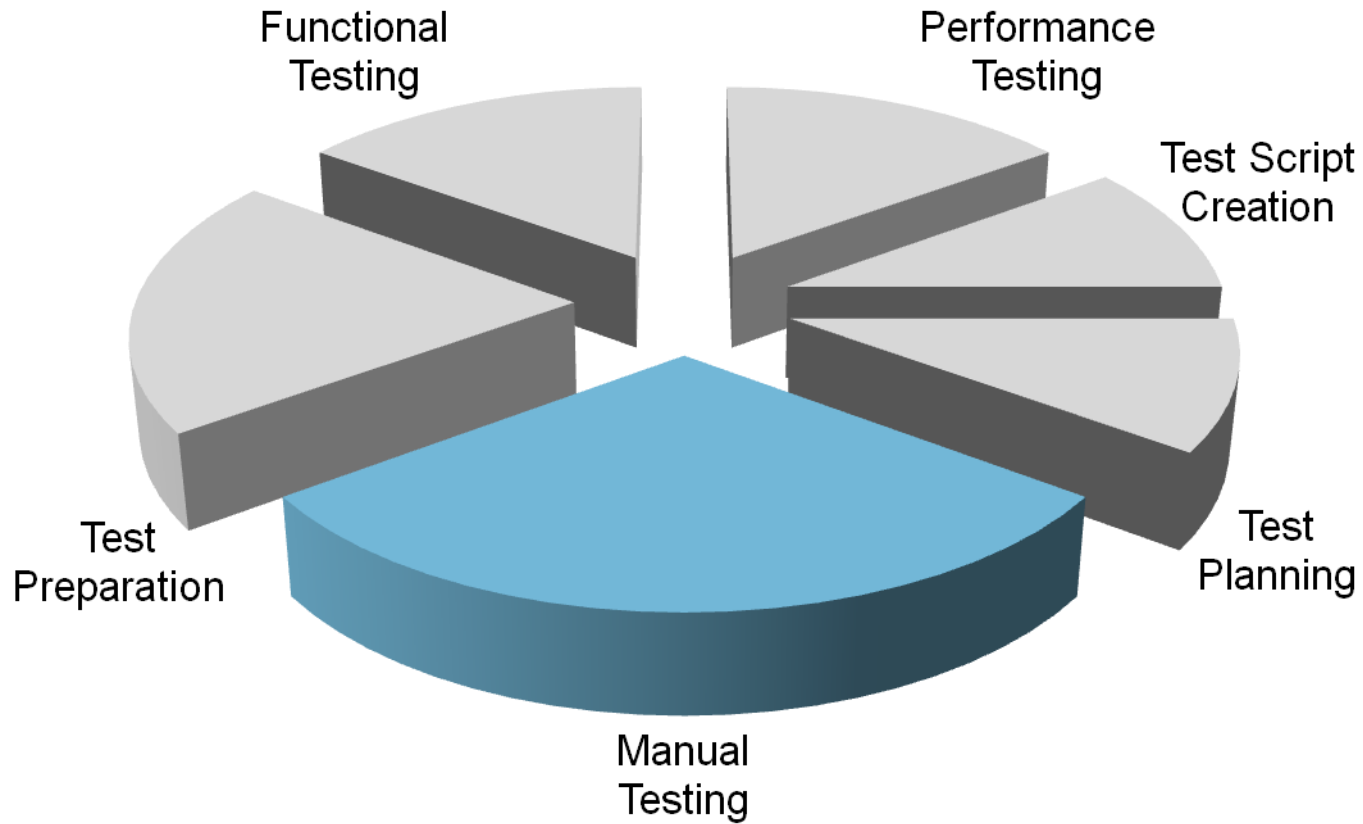


目标- 降低流程失效、维持进度

- 对Release的影响
  - 任何里程碑没有重大缺陷
  - 按时交付
  - 按计划内容交付

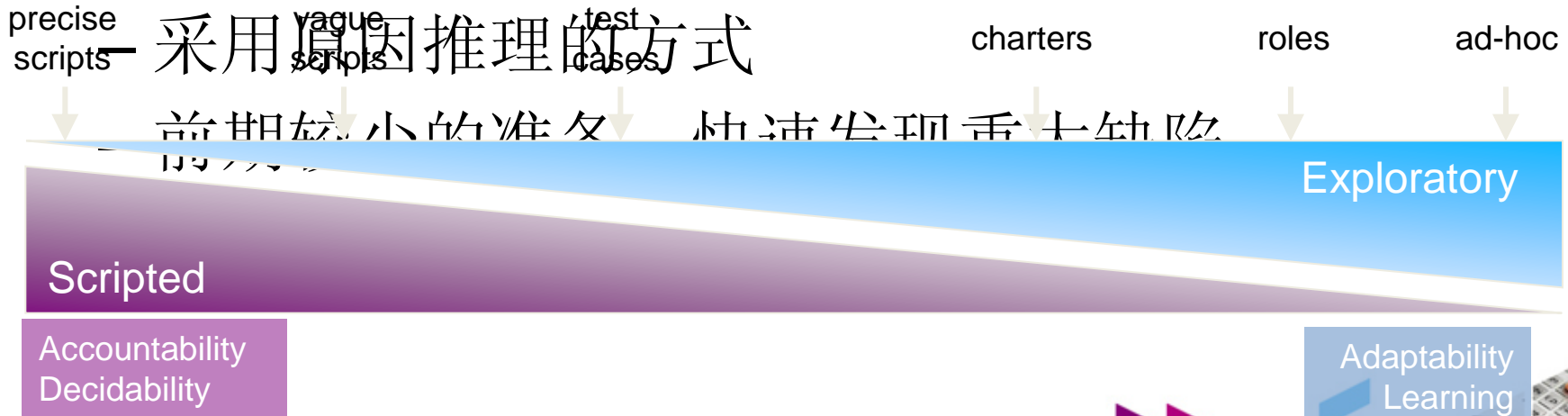


# 手工测试



# 手工测试的方式

- 脚本化的测试
  - 测试用例先被设计，并被记录下来
  - 测试用例可能被其他人在后期执行
- 探索性测试
  - 业务学习、测试设计和执行同时进行



# 适用于所有人员的测试描述语言

- 类自然语言的脚本描述
- 使得业务分析人员，UI设计人员，开发人员以及其他相关人员能够共享统一的脚本语言



捕获业务流程



定义功能需求

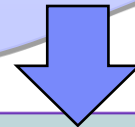


描述工作场景



评估易用性

1. Go to <http://demo.testfire.net/>
2. Verify there is a "Sign In" link
3. Click "Sign In" link
4. Enter "jsmith" into the "Username:" textbox
5. Enter "demo1234" into the "Password:" textbox
6. Click "Login" button



PERSONAL

## Online Banking Login

Username:

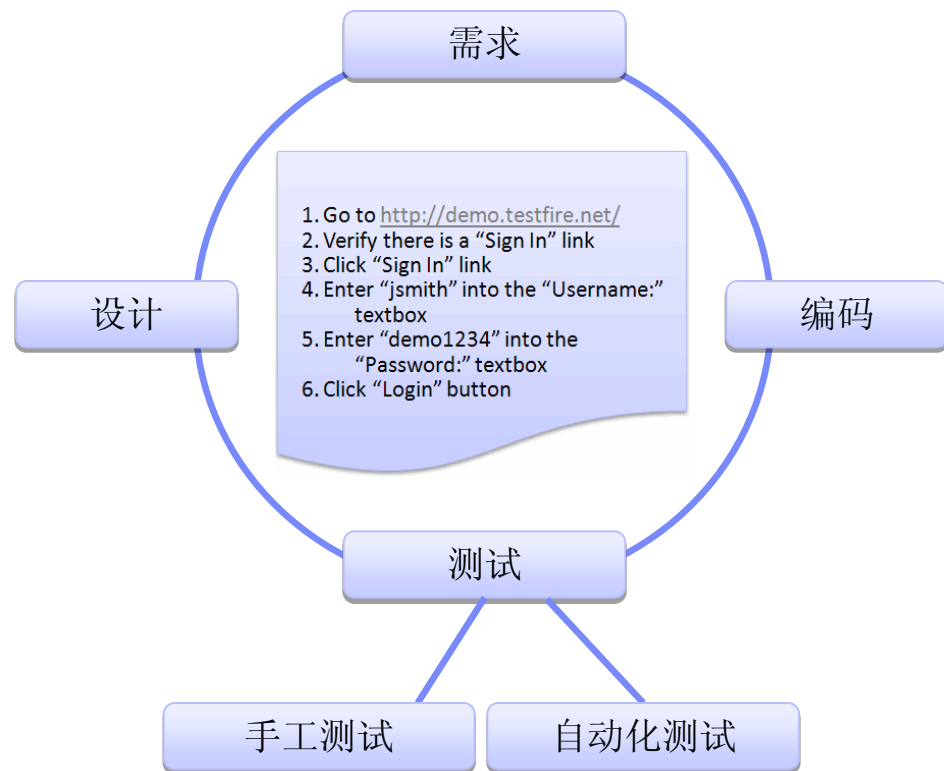
Password:

Login



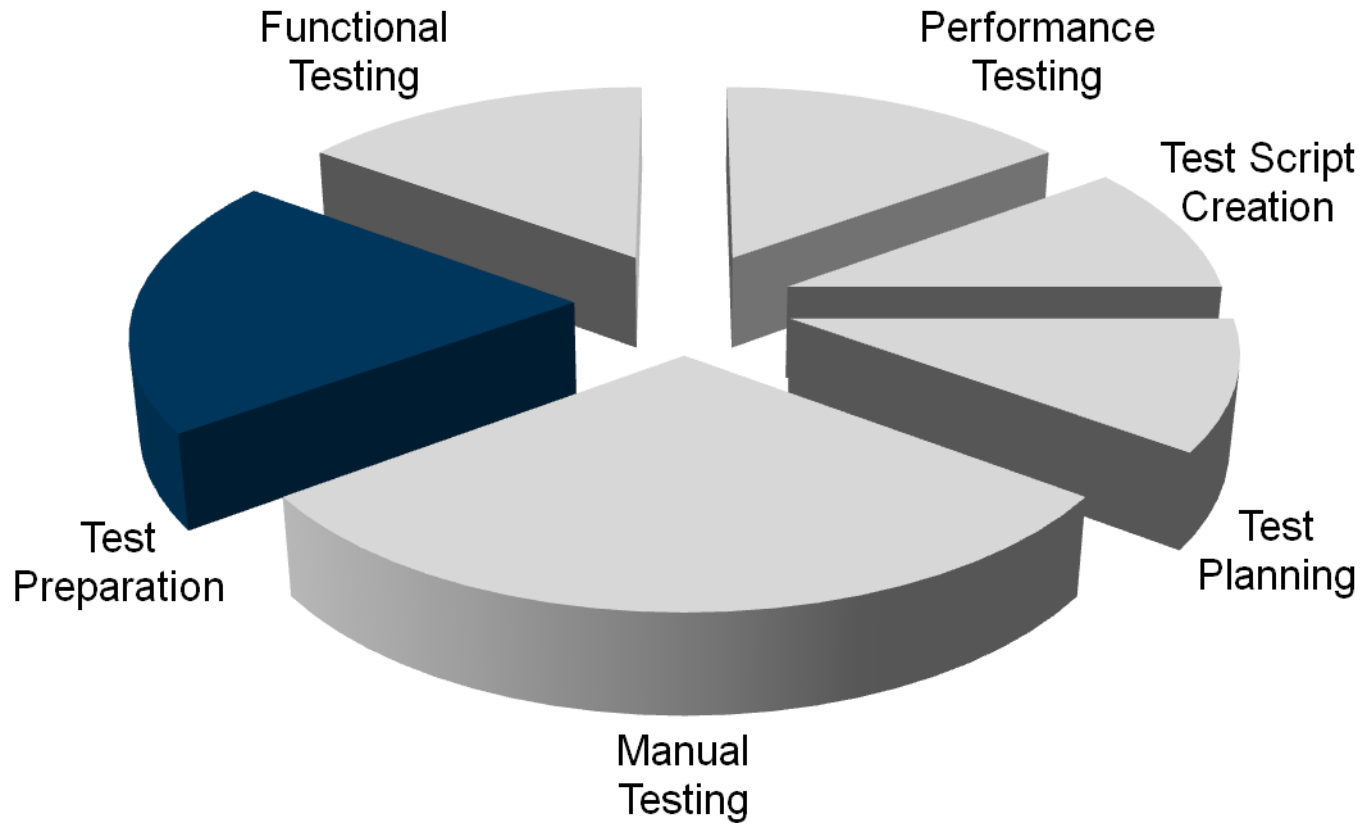
# 一次描述，广泛应用于质量生命周期

- 在开发的早期捕获测试用例
- 在系统细化过程中，作为手工测试脚本
- 当代码稳定后，转换为自动化回归测试脚本



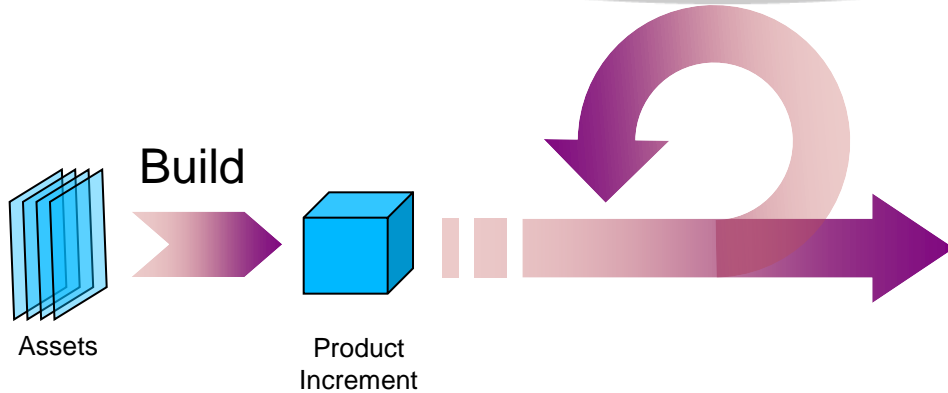
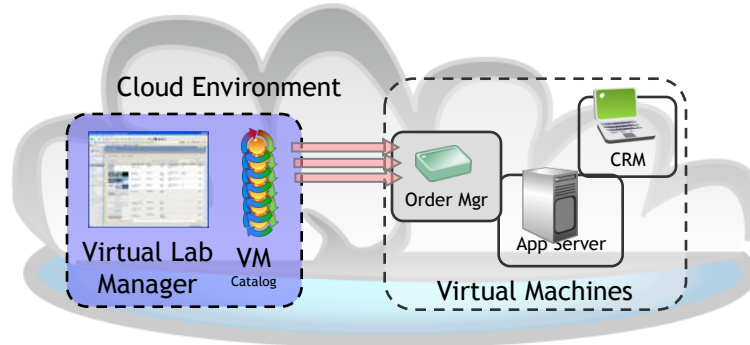


# 测试准备



# 测试环境自动化

实现编译，部署，发布，冒烟测试和自服务的  
sandboxes的自动化



- 冒烟测试
- Sandboxes
- 自动安装

# 灵活的、成本可控的部署模式



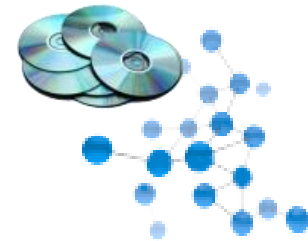
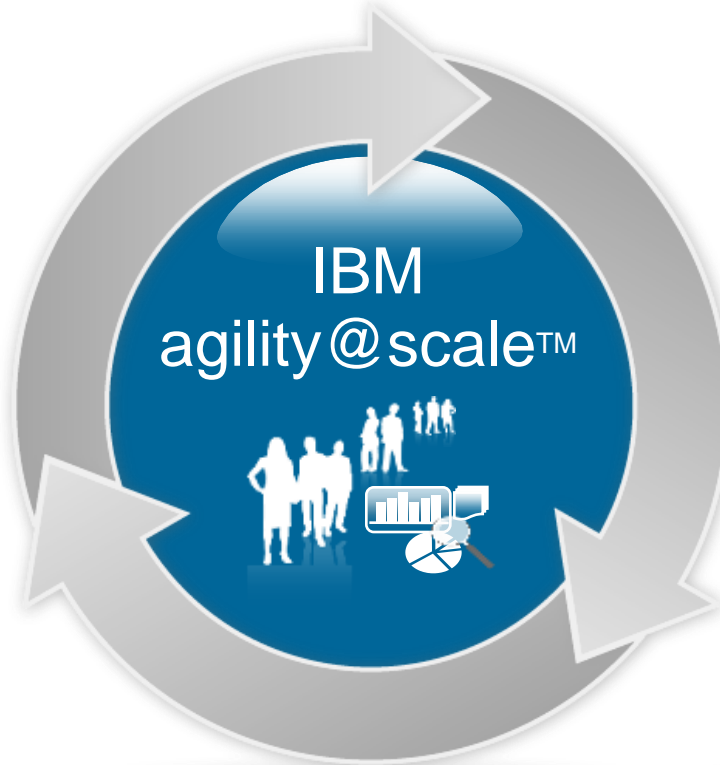
Cloud Provisioning



WebSphere, Tivoli, Lotus, Rational, Information Management



Integrated Infrastructure "Stacks"



Rational Tools virtual images

# 内容

- 敏捷质量
- IBM敏捷质量战略
- 小结





# 小结

- IBM 敏捷质量战略
  - ▶ 关注团队以及RQM和Jazz产品的协作
  - ▶ 借助合作伙伴扩大测试范围
  - ▶ 基于过程数据的收集和分析, 提升测试和质量的ROI
  - ▶ 引入探索性测试, 并提供探索性测试的自动化之路
  - ▶ 将质量延伸到部署自动化



# 小结

- 敏捷质量管理的最佳实践
  - ▶ 期待并接受变化
  - ▶ 通过快速响应和轻量的文档降低变更的成本
  - ▶ 提供在线实时的状态和进度报告，以便与变更的通知
  - ▶ 在会议过程中收集确认测试用例，提升会议质量
  - ▶ 测试作为完成的标志





