



IBM Software Group

最新IBM测试管理解决方案



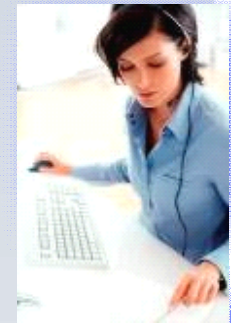
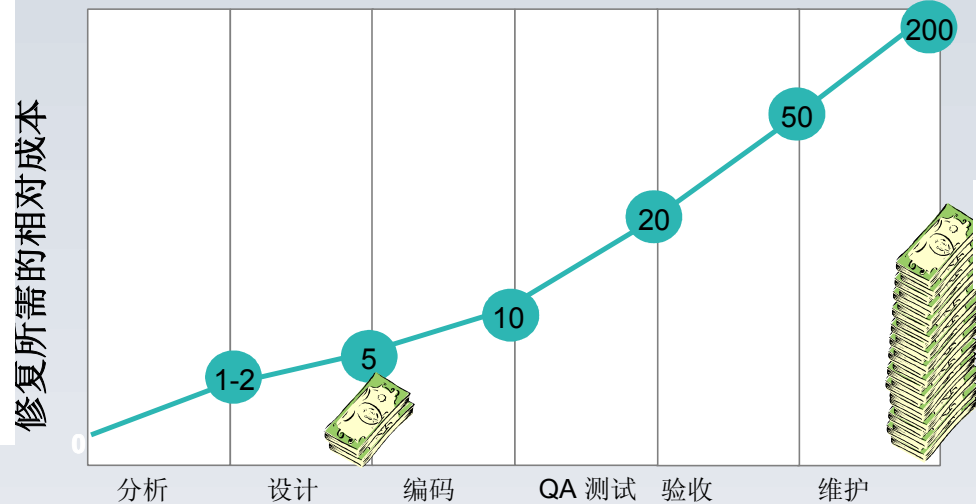
是否我们的注意力集中在需求和质量上？

传统方法导致大量返工、工期延迟和低下的质量

丢失的机会

- 推迟上市 **6** 个月以上将导致组织损失 **5** 年 ROI 的 **33%**
- **41%** 的项目未能交付预期的业务 ROI
- **49%** 的项目超出了最初的预算

Standish Group



80% 的开发成本都被用来识别和纠正缺陷！

传统 QA 测试

- **25 – 30%** 的交付时间都用在测试中 (IBM Study)
- 糟糕的上游质量导致了返工
- 被压缩的计划使这种情况变得更糟



问题：孤立的开发生命周期

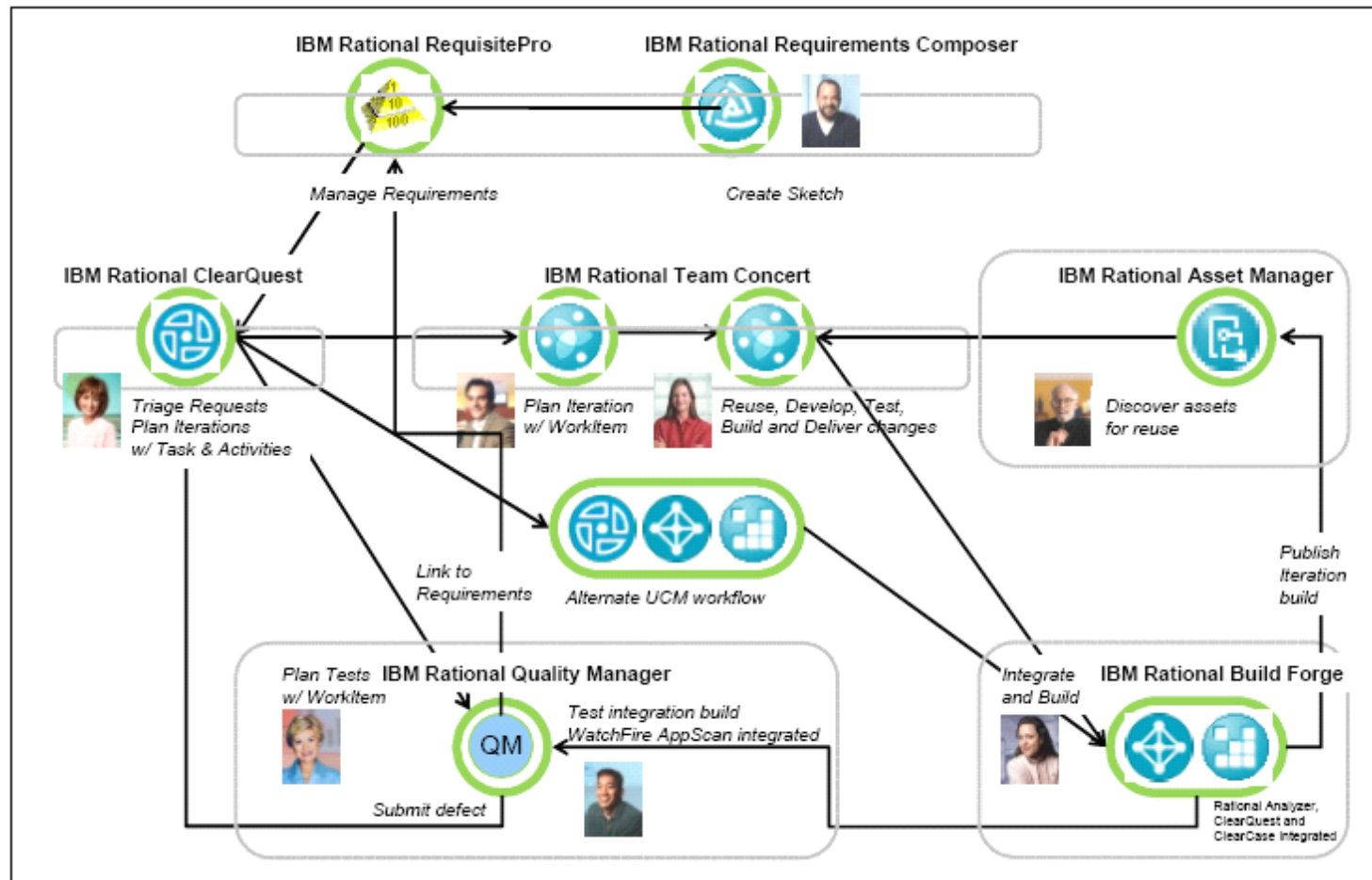


- 需求经常丢失 – 未能满足涉众需求和业务需求，缺少计划和成本目标
- 缺乏可追踪性 – 无法证实遵从性
- 缺乏影响分析 – 无法反映不断变化的业务需求
- 对“感知的”需求进行测试 – 验收测试失败之后返工，导致计划变动和更高的成本
- 业务的变化不能被追踪到测试 – 由于高优先级的需求可能被放到流程末尾并被丢弃，导致更高的业务风险

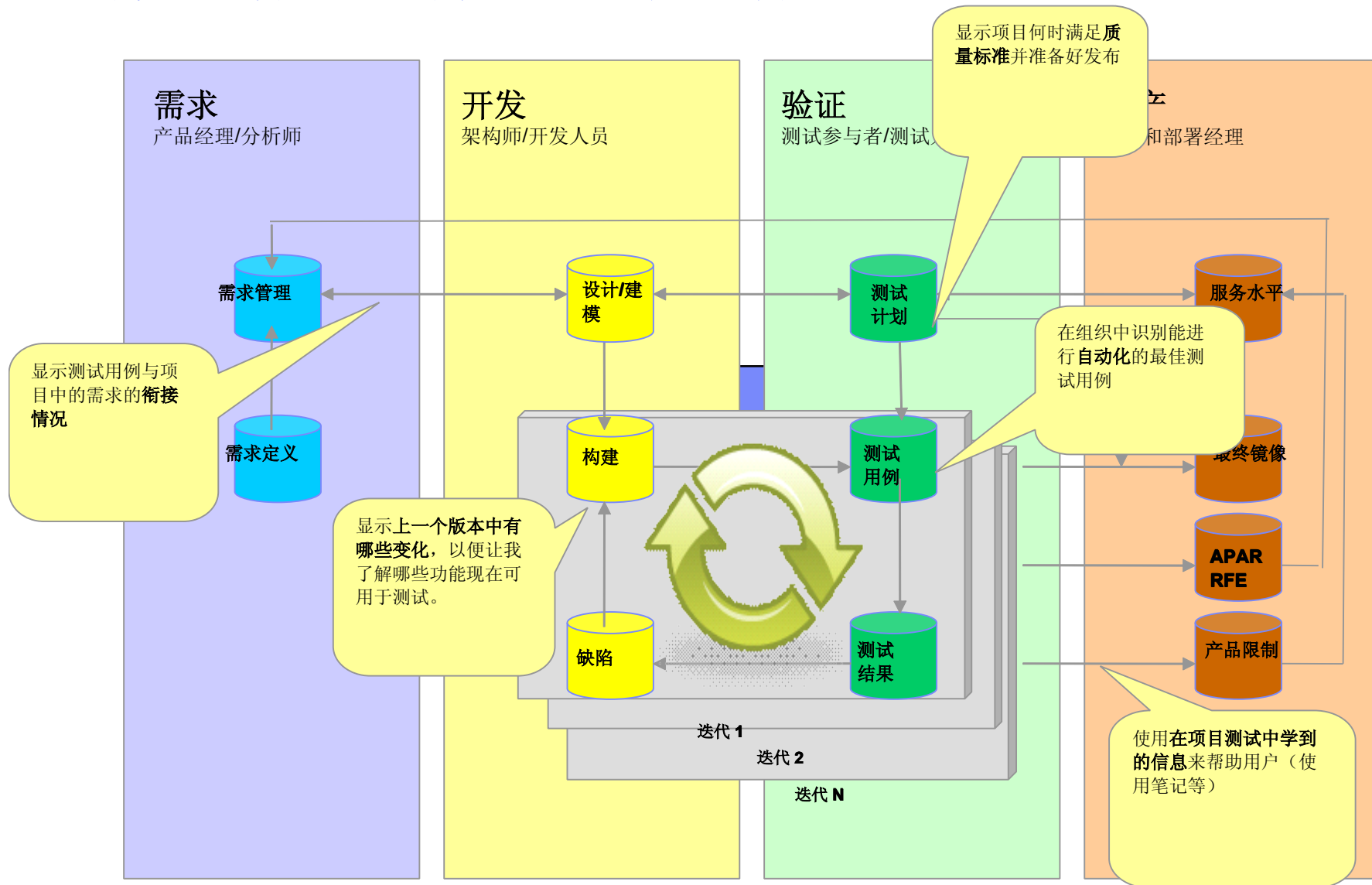


那么我如何修复这些问题？而且不再延期和超出预算？

跨越应用质量的开发生命周期



延伸：跨越应用质量生命周期



议程

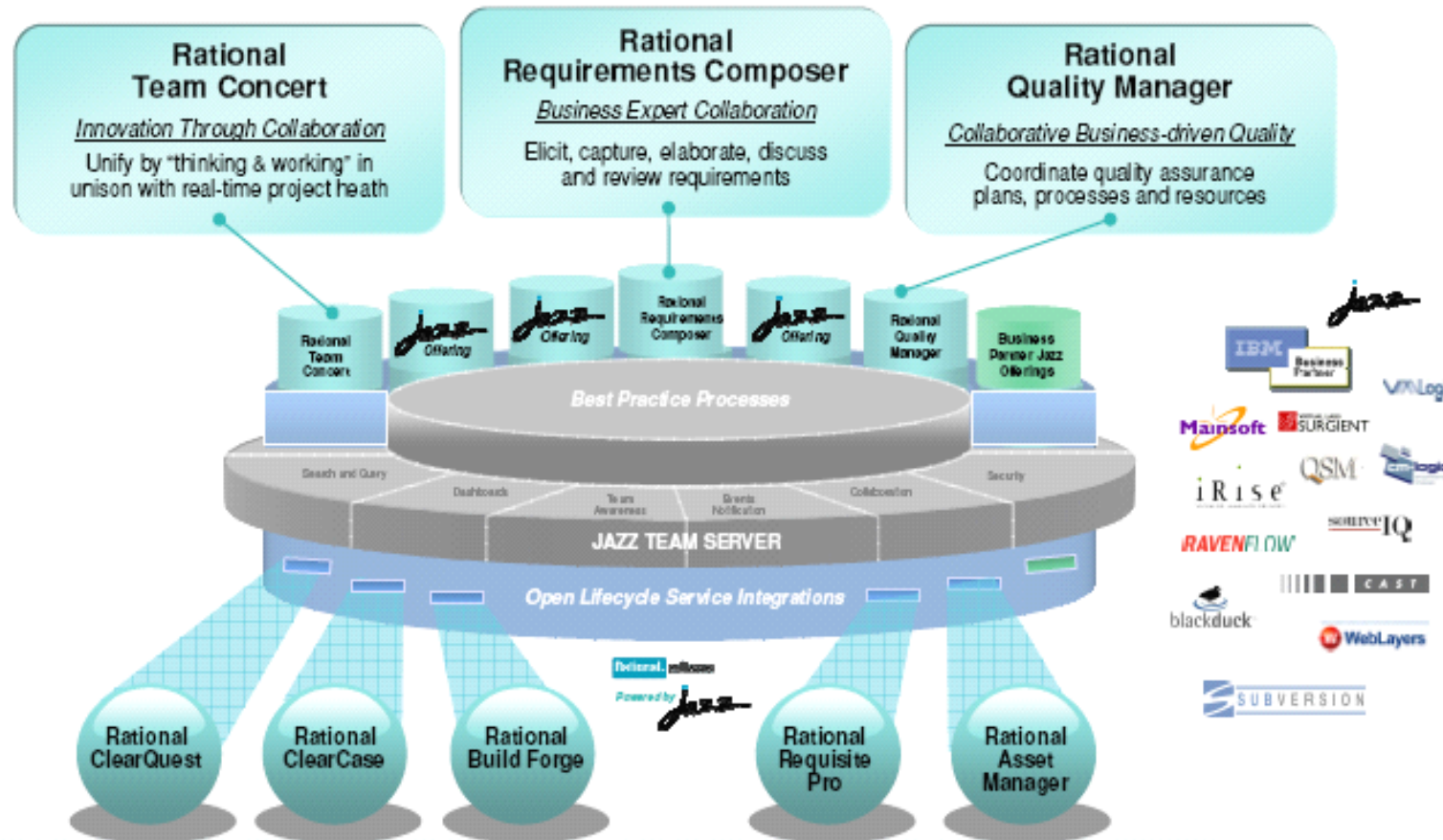
- 质量管理工具
- 功能测试工具
- 性能测试工具



保证业务质量



基于Jazz的质量管理解决方案--RQM



IBM Rational提升质量管理

减少业务风险



协作

协作的、持续的和全面的信息共享，减少缺陷、改进传递效率并增加客户满意度

改进运行效率



自动化

使劳动密集型活动自动化，缩短上市时间、增加可预见性和一致性，改进投资回报

作出自信的决定



报告

时刻了解软件质量指标，为发布作出决策，以满足业务和IT需要



使用集中式的Hub来管理业务驱动的质量

通过协作降低业务风险

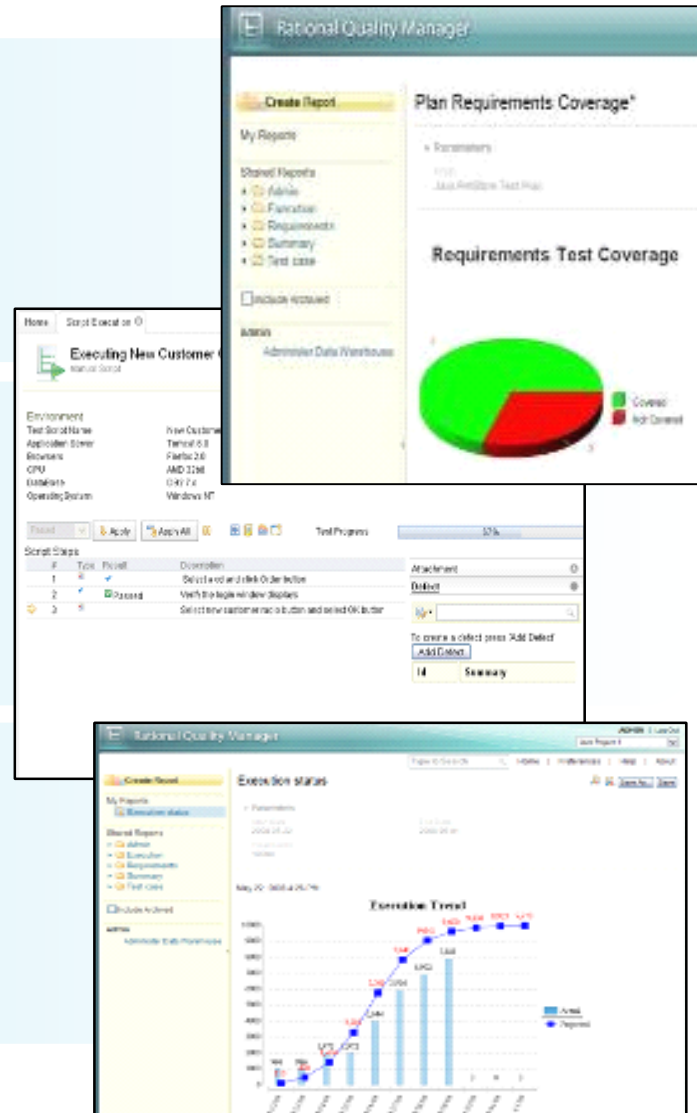
- 在团队中协调测试计划
- 可实施的过程 workflow
- 上游和下游质量

通过自动化改进运营效率

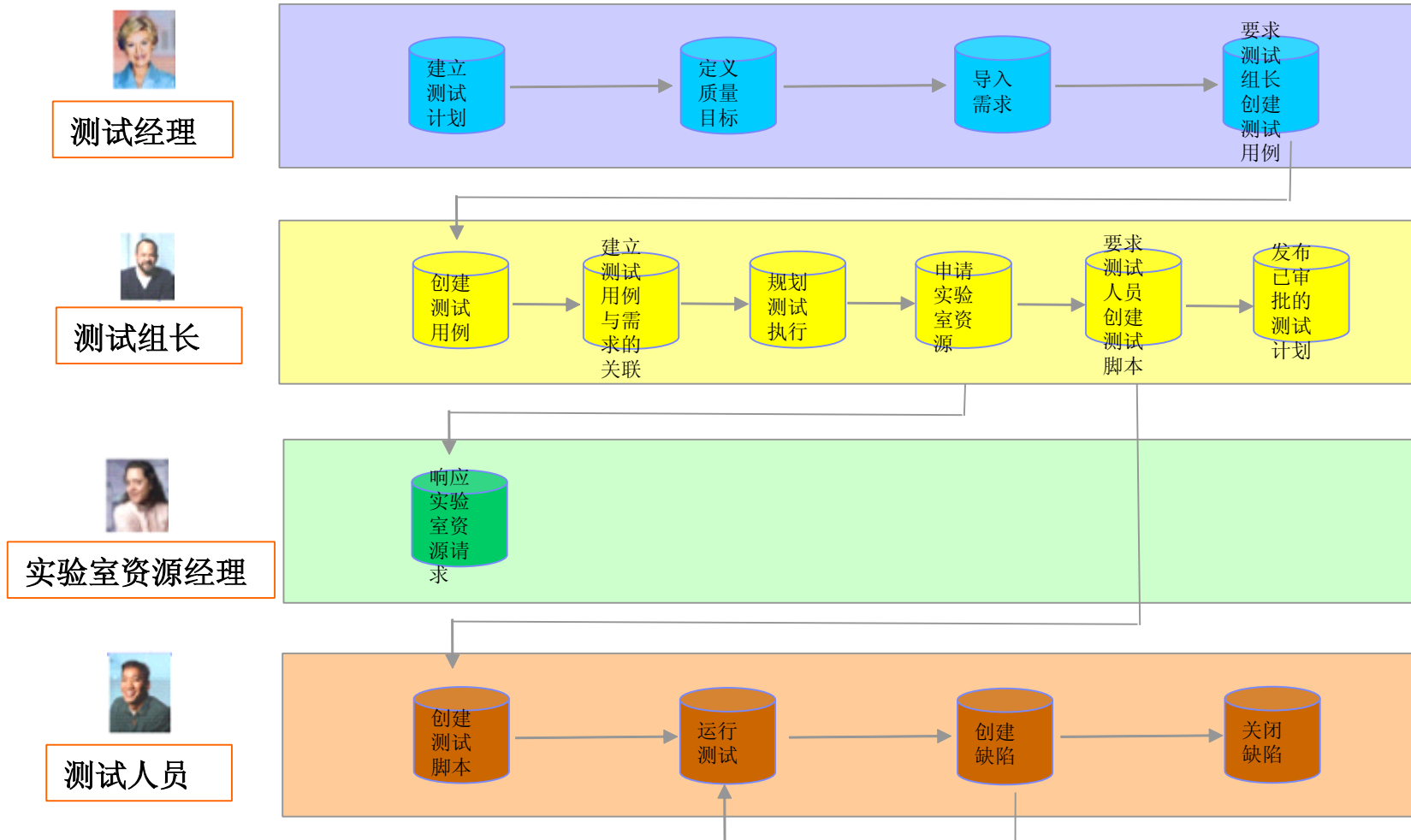
- 提高实验室效率和资产利用率
- 测试配置优化
- 环境和生命周期覆盖

通过快速的报告作出自信的决定

- 动态分析和流程改进
- 主动的风险预防
- 更高的可预测性



基于角色的测试 workflow



测试经理的测试 workflow



测试经理

建立
测试
计划

定义
质量
目标

导入
需求

要求
测试
组长
创建
测试
用例



质量仪表盘

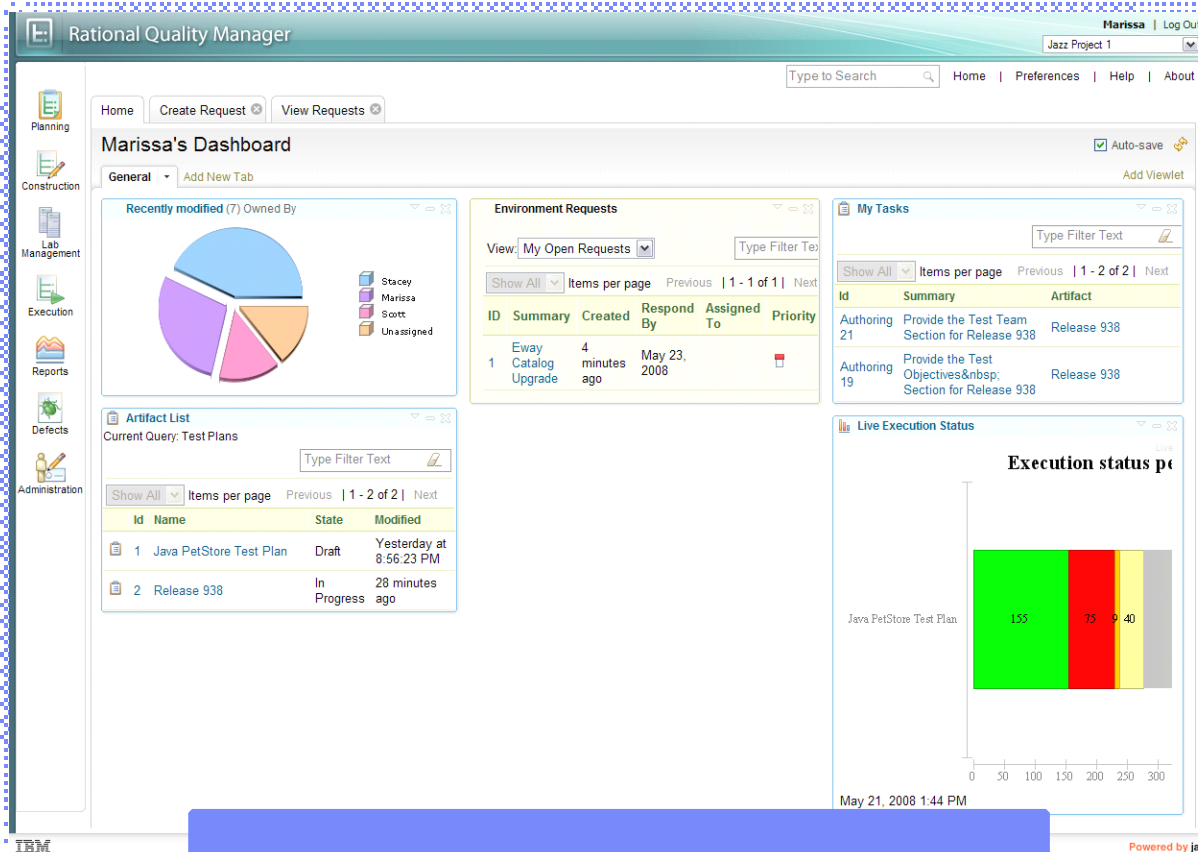
The screenshot shows the Rational Quality Manager interface. The top navigation bar includes 'Rational Quality Manager' and a 'Trial License expires' notice. The main area is titled 'Mary (Manager)'s Dashboard' and contains several viewlets:

- Work Item Queries:** A list of items assigned to the user, including '7: Author Test Case Security Login Test 7', '29: ADMIN Test Plan Summary', '30: Mary, please provide the information', '32: Author Test Case ADMIN_Test_Plan', '33: Author Test Case ADMIN_Test_Plan', '38: Requirement AMY ADDED 02', and '45: PetStore Test Plan - Please Mary add Summary session of PetStore Test Plan'.
- My Tasks:** A table listing tasks with columns for Id, Summary, and Artifact.

Id	Summary	Artifact
45	PetStore Test Plan - Please Mary add Summary session of PetStore Test Plan	Java PetStore Test Plan
33	Author Test Case ADMIN_Test_Plan	ADMIN_Test_Plan
32	Author Test Case ADMIN_Test_Plan	ADMIN_Test_Plan
30	Mary, please provide the information	ADMIN_Test_Plan
29	ADMIN Test Plan Summary	ADMIN_Test_Plan
7	Author Test Case Security Login Test 7	Security Login Test 7
44	Approve/Review the artifact	Test case
	Approve/Review the	ADMIN_Test_Plan
- Requests:** A viewlet showing 'All Assigned to Me' with a table header including ID, Summary, Created, Respond By, Assigned To, and Priority. It currently displays 'No items found.'
- Reservations:** A viewlet showing 'My Active Reservations' with a table header including ID, Machine, Start Date, End Date, Request ID, and Request Status. It currently displays 'No items found.'

Callouts in the image point to the 'General' tab, the 'Add New Tab' button, and the 'Add Viewlet' button.

测试经理的仪表盘 被定制的信息实时刷新



使用者可自己定义版面和内容



测试计划管理

支持迭代的测试计划

The screenshot displays the Rational Quality Manager web interface. The main content area shows a test plan titled "Java PetStore Test Plan" in a "Draft" state. A "Table of Contents" sidebar on the left lists various sections like Summary, Business Objectives, and Test Iterations. A "Summary" section is currently selected, showing the "Owned By" field set to "Mary (Manager)" and a "status" of "New".

Three blue callout boxes with white text and arrows point to specific features:

- 追踪测试计划历史** (Track test plan history) - points to the "View Snapshots" link.
- 指定属主** (Specify owner) - points to the "Owned By" dropdown menu.
- 使用多用户定义section, 构建测试计划** (Use multi-user defined sections to build test plan) - points to the "Table of Contents" sidebar.

为测试计划添加测试需求

Requirements ? ✕ ⌵

Owned By: ⌵ Status: ➔ ⌵ 📄 Authoring 1

This section lists all of the content and requirements associated with a given test plan. You can select existing requirements or define new items to cover in the test plan.

Group by: ⌵ 📄

⌵ Items per page Previous | 1 - 1 of 1 | Next 📄 📄 + 📄

	Status	Id	Name	Description	Owner
<input type="checkbox"/>	🟢	30	Java PetStore Shopping Req	Test the functionality of the PetStore shopping cart.	Mary (Manager)

Previous | 1 - 1 of 1 | Next

添加测试任务

Authoring 1

Summary: →

Overview | [Links](#) | [Approvals](#) | [History](#)

Loaded: Jul 1, 2008 10:25 AM

Details

Type:	<input type="text" value="Authoring"/>	Owned By:	<input type="text" value="Mary (Manager)"/>
Creation Date:	<input type="text" value="Jun 28, 2008 5:53 PM"/>	Estimate:	<input type="text" value=""/> Correction: <input type="text" value=""/>
Created By:	<input type="text" value="ADMIN"/>	Due Date:	<input type="text" value="Jun 29, 2008"/>
Project Area:	<input type="text" value="Samples"/>		
Team Area:	<input type="text" value="Lab Managers"/>		
Filed Against:	<input type="text" value="Samples"/>		
Tags:	<input type="text"/>		

Description [Edit](#)

Author Test Case Java PetStore Test Plan

Discussion (1 comment) [Add Comment](#)

[Collapse All](#) | [Expand All](#)

- [Amy \(Architect\)](#) Jun 30, 2008 1:35 PM
Mary -- just added a new requirement.

[Add Comment](#)



测试组长可看到分配给自己的任务

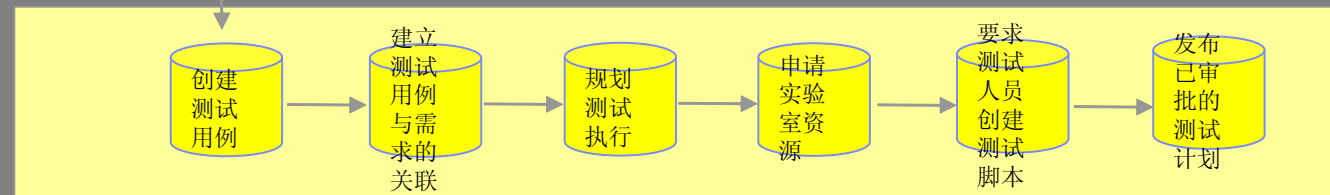
The screenshot shows the Rational Quality Manager interface for a user named Mary. The main content area is titled "Mary's (Manager) Dashboard" and contains several panels. The "My Tasks" panel is a table listing tasks assigned to the user. One task, with ID 1, is circled in red. The "Requests" panel shows a list of requests, including one from "bianca's request 1".

Id	Summary	Artifact
9	Author Test Script Defect_Analysis	Defect_Analysis
7	Author Test Case Security Login Test 7	Security Login Test 7
1	Author Test Case Java PetStore Test Plan	Java PetStore Test Plan
22	Author Test Approve new the artifact	Java PetStore Test Plan

测试组长的测试 workflow



测试组长



测试用例

Rational Quality Manager

Home | New Test Case | All Test Plans | ADMIN_Test_Plan | Test case * | New Test Case

Table Of Contents

- Summary
- Test Case Design
- Review and Approvals
- Requirements
- Pre-Condition**
- Post-Condition
- Expected Results
- Test Scripts
- Execution Work Items**
- Attachments
- Show All Sections

< Enter New Test Case Name > ? New

Test Case Overview | View Snapshots

Originator: ADMIN State:

Summary ?

Owned By: Status:

Use the theme, category and function features to group your test cases along related items or logical groupings. Weight is based on tester hours, units of work, or priority.

Theme:

Category:

Function:

Weight: Points

Description:

[< Add Content >](#)

为测试关联需求

Test Cases ? ✕ ⌵

Owned By: Status: ➔ 📄 Authoring 1

Lists the test cases associated with a given plan. You can add and remove associations to test documents and create and associate a new test case. Removing a test case will remove the association to this test plan but not delete the test case.

Group by: 📝

Previous | 1 - 4 of 4 | Next 📄 + 📄 📄

		Id	Name	Owner	State	Theme	Category	Function	Weight	Modified
<input type="checkbox"/>		6	Reporting Dashboard Test 6	Donald (Developer)	Draft	Performance	Reporting	Dashboard	70	Yesterday at 7:24:28 PM
<input type="checkbox"/>		7	Web Services Login Test 7	Laura (Lab Manager)	Draft	Usability	Web Services	Login	60	Yesterday at 7:24:30 PM
<input type="checkbox"/>		8	Rich Client UI Execution Test 8	ADMIN	Draft	Functionality	Rich Client UI	Execution	55	Yesterday at 7:24:32 PM
<input type="checkbox"/>		10	Java PetStore Shopping Test	Unassigned	Draft	Functionality	Web UI	Execution	5	1 minute ago

Previous | 1 - 4 of 4 | Next

规划测试执行

Rational Quality Manager

Home All Test Plans Classics Admin

Create Section

Table Of Contents

- Summary
- Business Objectives
- Test Objectives
- Review and Approvals
- Requirements
- Application Security
- Test Iterations
- Sizing
- Environments
- Test Team
- Quality Goals
- Entry Criteria
- Exit Criteria
- Test Cases
- Attachments
- Show All Sections

Classics Admin
Test Plan Overview | View Snapshots

Discard Changes Save

Originator: Mary (Manager) State: Draft

this test plan covers the scope and approach for testing the administrative functions of Classics CD

Environments

Owned By: Unassigned Status: New

Lists the various environments supported and tested by this test plan. You can add various platforms such as browsers, databases, operating systems and other items. This list is then used to generate test configurations.

Type Filter Text

Show All Items per page Previous | 1 - 5 of 5 | Next

Type	Attributes
Application Server	Tomcat 6.0, WAS 6.1, WAS 7.0,
Browsers	Firefox 2.0, Internet Explorer 6, Internet Explorer 7,
CPU	AMD 32bit, System i, System z,
DataBase	DB2, DB2 7.x, DB2 8.x,
OperatingSystem	Vista, Windows NT, Windows Server 2003,

- 指定测试用例的执行环境，并确定测试配置
- 测试配置常用来帮助搜索是否有可用于测试的机器
- 如果没有相应的测试环境，测试组长可提交购买请求

过程审核

对测试内容进行审核和审批

- 测试经理、测试组长review、refine 和sign-off所有和质量相关的工件
 - ▶ 需求、测试计划、测试用例、完成标准等
- 维护版本历史，追踪质量的演进

工件的审核和审批

Review and Approvals ?

Work Item: [Create](#)

List the people who will be reviewers and approvers of this content and define your approval process.

View: **All** Type Filter Text

Show All Items per page Previous | 1 - 2 of 2 | Next

	Review Type	Name	Status	Comments	Modified
<input type="checkbox"/>	Approver	Mary (Manager)	Pending		
<input type="checkbox"/>	Reviewer	Larry (Lab Manager)	Pending		

工件版本

Java PetStore Test Plan ?

Test Plan Overview | View Snapshots

Saved successfully at: 20:47:42 [Create New Snapshot](#)

Type Filter Text

Show All Items per page Previous | 1 - 1 of 1 | Next

Type	Revision	Name	Date	Originator	Details	Copy
	1	Draft version	1 minute ago	ADMIN		

Previous | 1 - 1 of 1 | Next

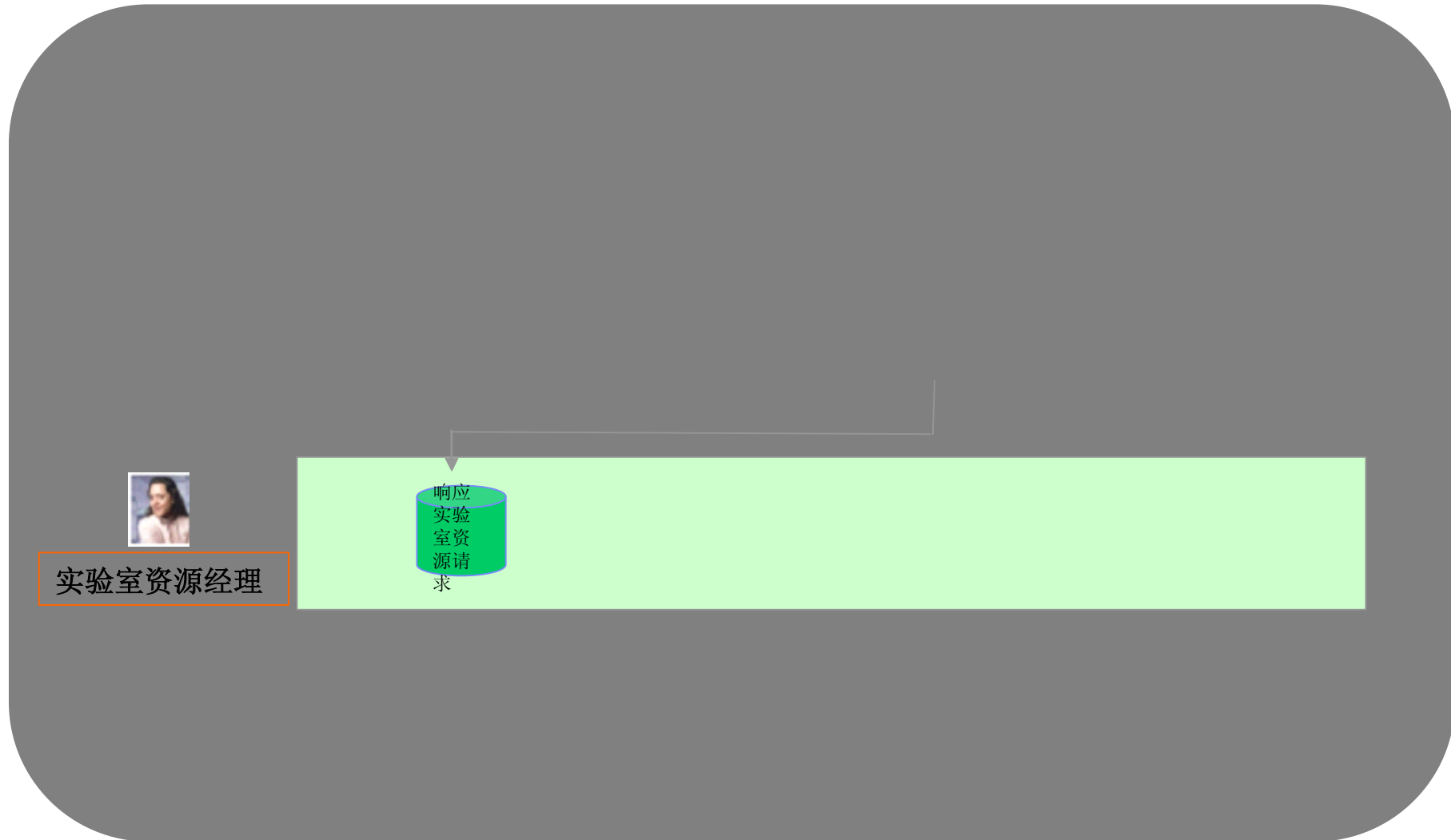
申请测试环境资源

The screenshot displays the Rational Quality Manager web interface. The main content area shows a 'Request <11:21:07>' form. The 'Summary' field contains 'Administration Scenario Test Request'. The 'Reservation' section includes 'Reserve From' (6/9/08, 8:00 AM) and 'Reserve To' (6/13/08, 3:00 PM). The 'Configuration' section has a text input field for 'Enter a label for the Lab Resource Description' and a 'Less...' button. Below this, system specifications are listed: Processor Type (AMD 32bit), Installed Software (Firefox 2.0, DB2 7.x, Tomcat 6.0), and Operating System (Windows NT).

- 提交请求，要求实验室资源经理提供测试环境
- 包含两个主要内容：使用环境的期限和环境的配置



实验室资源经理的测试 workflow



查找可用的测试资产

The screenshot displays the Rational Quality Manager web interface. The top navigation bar includes 'Home', 'All Test Plans', 'All Requests', and 'View Request'. The main content area shows a search for machines with the status 'Searching for machines to fulfill request...'. A 'Configuration' section is visible, containing a search input field and a table of installed software. The table lists Processor Type (AMD 32bit), Operating System (Windows NT), and three installed software packages: Firefox 2.0, 7.x, and Tomcat 6.0. A 'Machine' table below shows a single entry 'demosys' which is checked. Navigation controls for the machine list include 'Show All', 'Items per page', and 'Previous | 1 - 1 of 1 | Next'. A 'Reserve' button is located at the bottom right of the machine list.

Processor Type:	Installed Software:	Installed Software:	Installed Software:
AMD 32bit	Firefox 2.0	7.x	DB2 Tomcat 6.0

Machine
<input checked="" type="checkbox"/> demosys

- 实验室资源经理从申请列表中查找新的请求
- 根据请求的测试配置，实验室资源经理查找可用的机器
- 如找到匹配机器，将被保留，供测试使用

执行自动化部署任务

Help

Automation

Automations
▶ ⚙️ ⚡ Less...

Available Jobs to Run

Previous | 1 - 3 of 3 | Next

Available Machines for Automations

Previous | 1 - 5 of 5 | Next

Name	Provider
<input type="checkbox"/> Check Dir	BuildForge RTLM Integration Adaptor
<input checked="" type="checkbox"/> Test Project	BuildForge RTLM Integration Adaptor
<input type="checkbox"/> RTLMTTestProject	BuildForge RTLM Integration Adaptor

Name
<input type="checkbox"/> RTLM IBM-C7E4BDD3EB7
<input checked="" type="checkbox"/> gwin318
<input type="checkbox"/> gvmw165
<input type="checkbox"/> gvmw150
<input type="checkbox"/> RTLM_TrainWreck

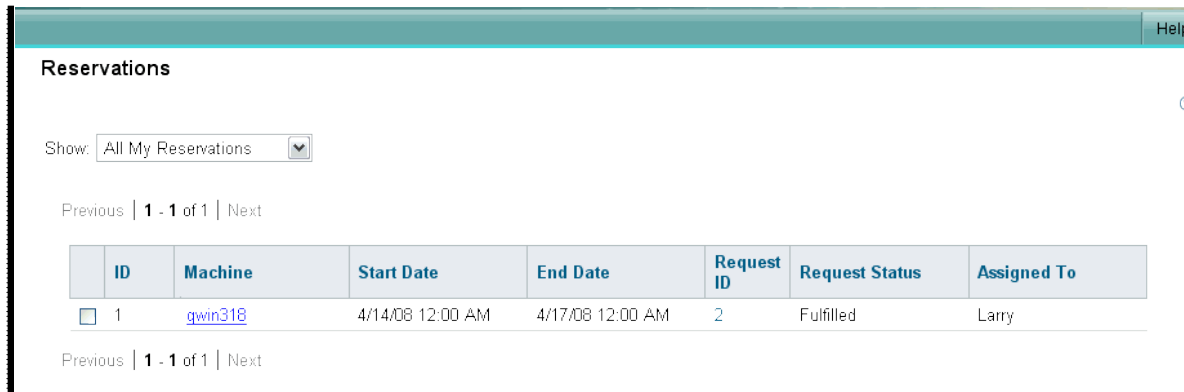
Running Automations
⚡ ⚡ Less...

Previous | 1 - 2 of 2 | Next

Job Name	Start Time	End Time
Test Project	12/31/69 7:00 PM	4/7/08 6:28 PM
RTLMTTestProject	4/7/08 4:53 PM	4/7/08 4:53 PM



通知申请者

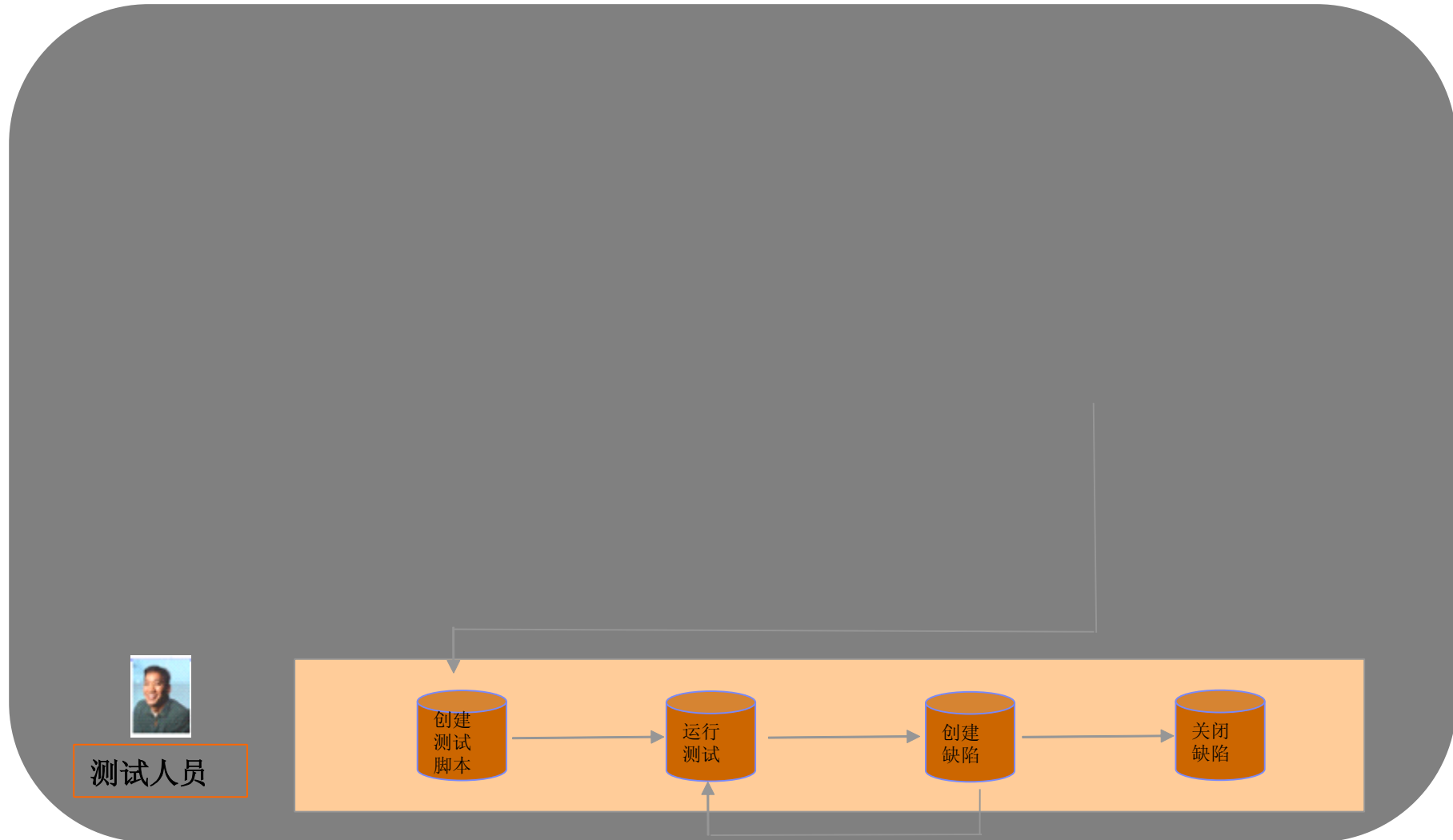


The screenshot displays a web application interface for managing reservations. At the top right, there is a 'Help' button. The main heading is 'Reservations'. Below the heading, there is a 'Show:' dropdown menu set to 'All My Reservations'. Navigation links 'Previous | 1 - 1 of 1 | Next' are present above and below the table. The table has the following columns: ID, Machine, Start Date, End Date, Request ID, Request Status, and Assigned To. A single row is visible with the following data: ID: 1, Machine: gwin318, Start Date: 4/14/08 12:00 AM, End Date: 4/17/08 12:00 AM, Request ID: 2, Request Status: Fulfilled, Assigned To: Larry.

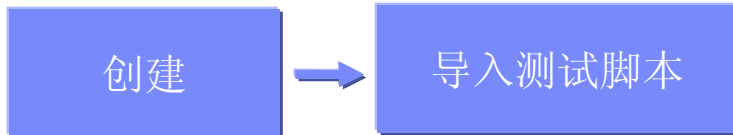
ID	Machine	Start Date	End Date	Request ID	Request Status	Assigned To
1	gwin318	4/14/08 12:00 AM	4/17/08 12:00 AM	2	Fulfilled	Larry

- 一旦系统被装好，实验室资源经理将通知申请者可进行测试
- 测试人员可开始进行测试

测试人员的测试 workflow



从外部测试工具导入自动化测试脚本



Import Test Script

Script Type: RFT Select Adapter

Project Path

Show Test Scripts

Current Query: Test Scripts to Import

Group by: Ungrouped

Type Filter Text

创建一个测试脚本的参考

为测试用例选择测试环境

Generate Execution Work Items: ✕

Step1 > Step2 > Step3

Generate Execution Work Items: Reporting Editors Test 2 Next >

Overview

Originator: Mary (Manager)

Owner: ▼

Test Plan: ▼

Iteration: ▼

Environment ?

Reuse Existing Configurations | Generate New Configurations

Select one or several attributes from the columns below. The number of work items created depends on : the number of attributes selected, the level of coverage that you choose, and the Advanced Properties (attribute Inclusions, Exclusions, and Weightings).

Application Server	Browsers	OperatingSystem
Tomcat <input type="checkbox"/>	<input checked="" type="checkbox"/> Firefox 2.0 <input type="checkbox"/> Internet Explorer 7	<input checked="" type="checkbox"/> Linux <input checked="" type="checkbox"/> Windows XP

[Advanced Properties](#)

Generation Properties ?

Coverage: ▼

Execution Work Items ?

Owned By: ▼ Status: ▼

Allows you to associate, create and generate execution items for an individual test case. Execution items contain detailed configuration information for the test case and the high level results from a test case execution. Detailed results are in the log files.

Group by: ▼ Type Filter Text

▼ Items per page Previous | 1 - 2 of 2 | Next

Name	Configuration	Test Plan	Iteration	Owner	Script	Status	Last Modified
<input type="checkbox"/> JavaPetStoreShoppingTest_Windows XP_Tomcat_Internet Explorer 7	Windows XP_Tomcat_Internet Explorer 7	Java PetStore Test Plan	M0	Tony (Tester)	<input type="text" value="Java"/> ▼ Java PetStore Shopping Script - None -	New	1 minute
<input type="checkbox"/> JavaPetStoreShoppingTest_Linux_Tomcat_Firefox 2.0	Linux_Tomcat_Firefox 2.0	Java PetStore Test Plan	M0	Tony (Tester)	<input type="text" value="Java"/> ▼ - None -	New	minute ago

Previous | 1 - 2 of 2 | Next

Run Script



执行测试并提交缺陷

Executing Security Editors Test 8 Manual Script Cancel

Environment
 Test Script Name: Security Editors Test 8 Script #1
 Browsers: Firefox
 CPU: x86
 OperatingSystem: Windows XP

Failed Test Progress: 100%

#	Type	Result	Description
1		Passed	Perform Step No 1
2		Passed	Perform Step No 2
3		Passed	Perform Step No 3
4		Passed	Perform Step No 4
5		Failed	Perform Step No 5

Defect 18 Save

Summary: * Wrong credit card format New Unresolved

Overview **Links** Approvals History Loaded: Jun 13, 2008 11:07 AM

Details

Type:	<input type="button" value="Defect"/>	Owned By:	<input type="button" value="Donald (Developer)"/>
Severity:	<input type="button" value="Blocker"/>	Priority:	<input type="button" value="High"/>
Found In:	<input type="button" value="Unassigned"/>	Planned For:	<input type="button" value="Unassigned"/>
Creation Date:	Jun 13, 2008 11:07 AM	Estimate:	<input type="text"/> Correction: <input type="text"/>
Created By:	ADMIN	Due Date:	<input type="button" value="Jan 19, 2008"/>
Project Area:	Samples		
Team Area:	Lab Managers		
Filed Against: *	<input type="button" value="Samples"/>		
Tags:	<input type="text" value="credit, card, java, petstore, shopping, cart"/>		

Description Edit

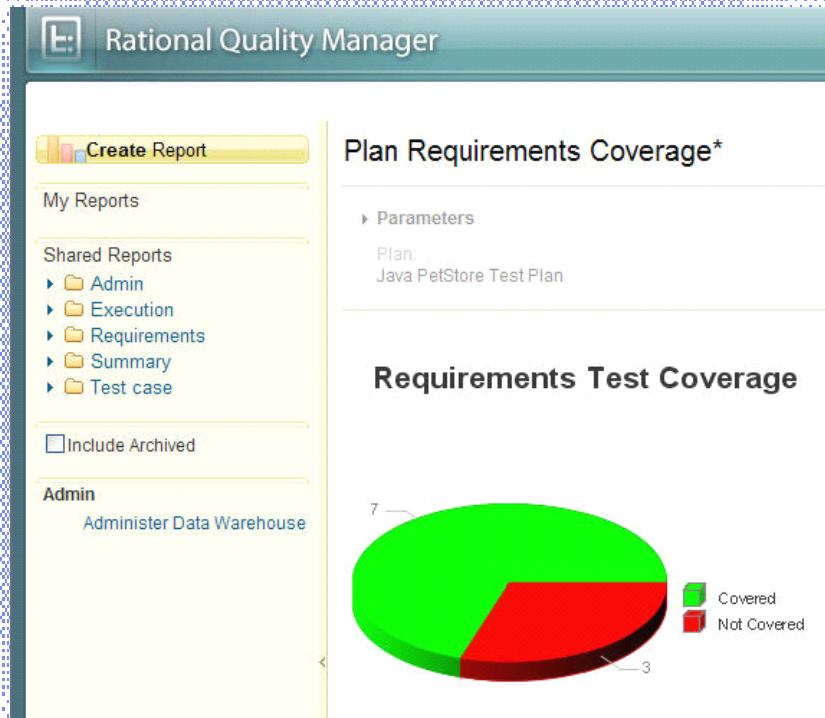
Entered 16-digit card number in credit card field. Format was not accepted.

Discussion Add Comment

No Comments. Add Comment

报表

来自多个角度的项目状态的静态视图



定制的报表能够共享和了解重要项目信息

议程

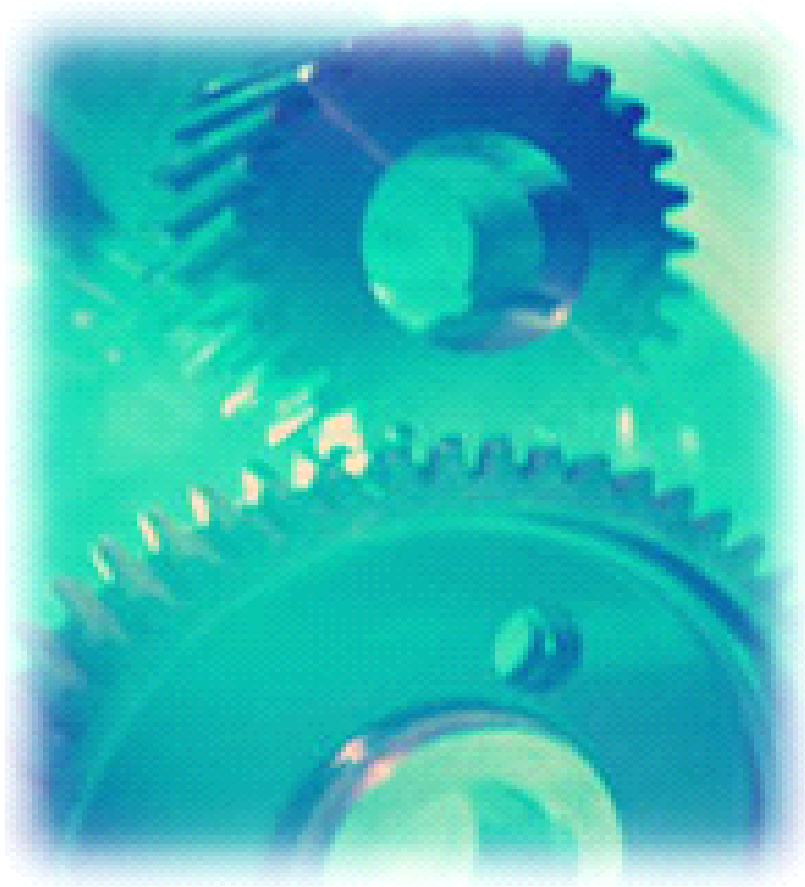
- 质量管理工具
- 功能测试工具
- 性能测试工具



保证业务质量



用 IBM Rational Functional Tester 进行功能测试 初学者和专业人员的自动化测试工具



- IBM Rational Functional Tester
 - ▶ WEB、J2EE和.Net系统的自动回归测试
- 主要优点
 - ▶ 降低功能测试成本
 - ▶ 快速生成测试脚本能力
 - ▶ 为高级测试人员提供强大的脚本语言和IDE

创建一个功能测试

只需要三个步骤



1、录制

- 脚本录制
 - ▶ 捕捉用户和系统在测试下的交互

2、修改

- 脚本修改
 - ▶ 捕捉用户和系统在测试下的交互

3、执行

- 脚本执行
 - ▶ 保证本地和远程执行的可靠回放



1、创建一个功能测试 录制脚本

- 创建测试脚本，捕获所有系统和用户的交互
 - ▶ 基于用户导航操作，生成脚本
 - ▶ 当要验证系统响应，插入验证点
- 脚本录制要考虑
 - ▶ 环境的支持
 - ▶ 创建数据驱动测试
 - ▶ 数据和属性的验证
 - ▶ 动态数据验证
 - ▶ 不能被自动化的内容



2、创建一个功能测试 修改脚本

- 基于录制的脚本修改代码
 - ▶ 增加VB.net或Java代码完成更多功能
 - ▶ 可能的修改：增加条件分支，数据池，重定义结构
- 修改脚本要考虑
 - ▶ 强大的编码语言
 - ▶ 专业的调试工具
 - ▶ 灵活的Object map编辑能力
 - ▶ 脚本的版本控制能力



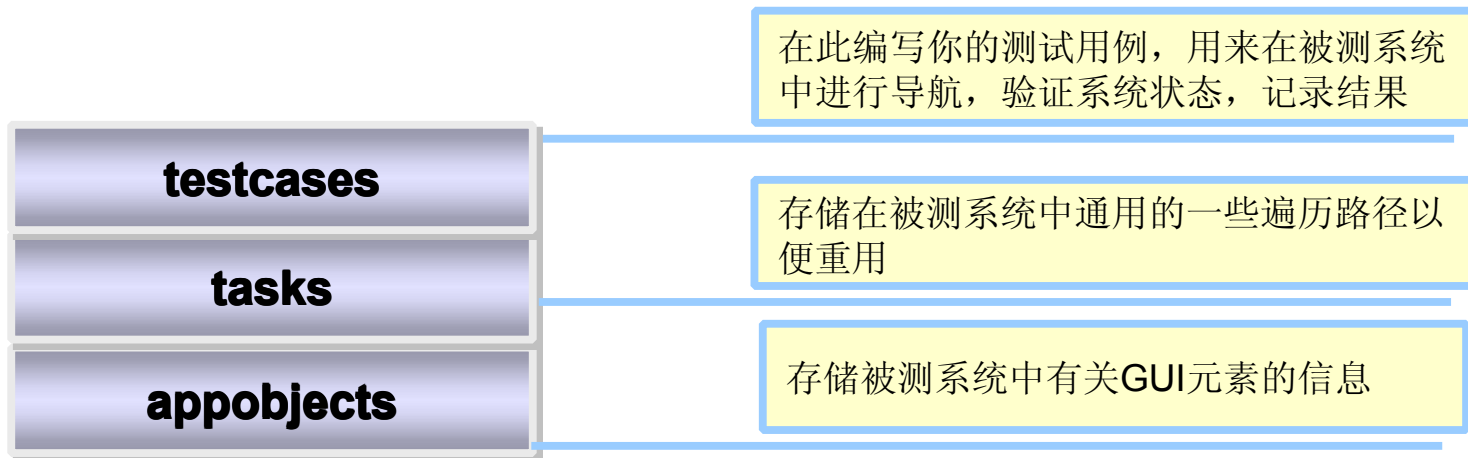
结合IBM自动化测试框架增强脚本编辑能力

- IBM自动化测试框架提供
 - ▶ 由appobjects, tasks和testcases组成的3层架构的实现
 - ▶ IBM类包
 - ▶ IBM最佳实践
 - ▶ 一个集成的IDE
- 它是用RFT进行GUI自动化测试的理论指导和最佳实践



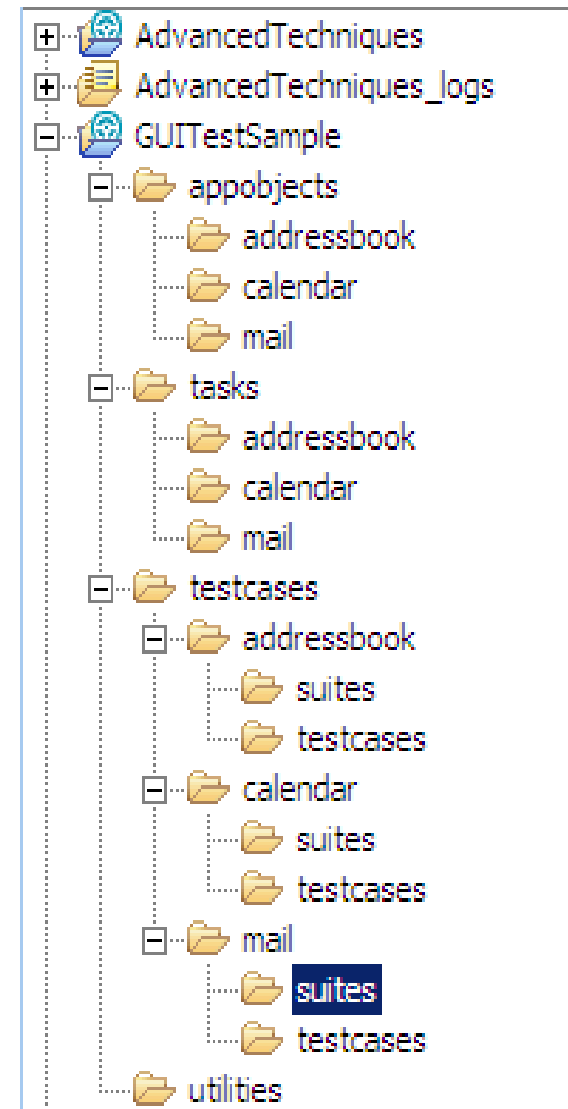
3层架构

- 由appobjects, tasks和testcases组成3层架构
- 分层的架构可以分离“做什么”和“怎么做”



如何组织3层架构

- Testcases层可以看作是你要测试“做什么”的一个高层的视图
- Tasks和appobjects层为testcases层提供必要的实现细节—测试“怎么做”



3、创建一个功能测试 执行测试

- 执行脚本，注意可能的差异
 - ▶ 执行脚本并产生测试日志
 - ▶ 测试日志会在实际和期望结果中高亮不同
- 执行脚本要考虑
 - ▶ 使用ScriptAssure技术确保可靠的回放
 - ▶ 可使远程和本地回放在不同平台进行



议程

- 质量管理工具
- 功能测试工具
- 性能测试工具



保证业务质量



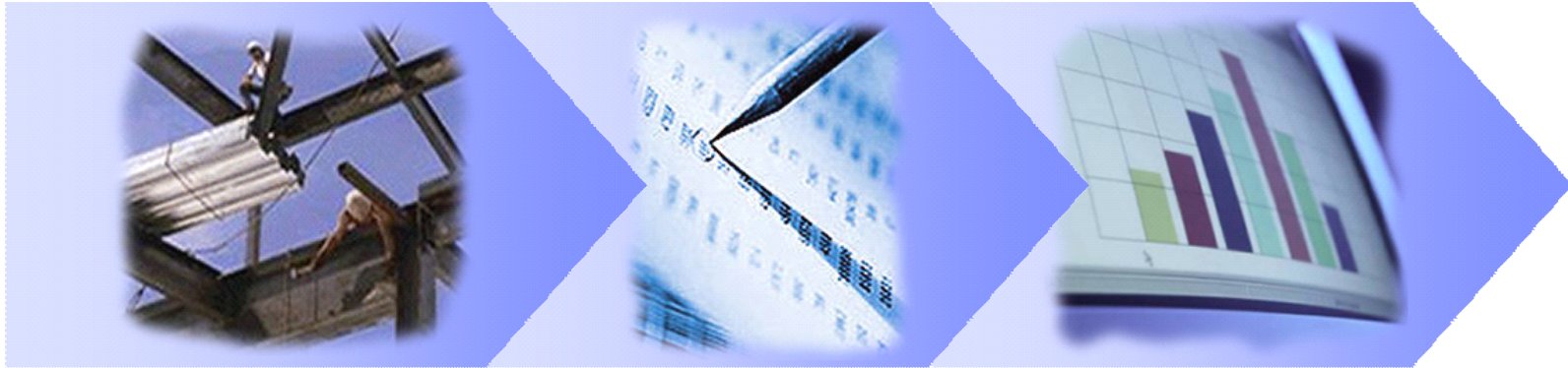
用 IBM Rational Performance Tester 进行性能测试 初学者和专业人员的自动化测试工具



- IBM Rational Performance Tester
 - ▶ 基于WEB系统的测试

- 主要优点
 - ▶ 保证工作生产力
 - 隐藏复杂度使工作完成
 - ▶ 先进的数据访问和处理
 - 数据变量及其同步
 - ▶ 低成本的性能测试工具
 - 低资源需要
 - 集成桌面开发工具

创建一个性能测试 需要3个步骤



1、建立脚本 2、制定工作负载(Workload) 3、执行和测试

- 脚本创建时需要考虑的因素
 - ▶ 可视化测试编辑器, 变量化输入数据, 关联服务器响应
- 工作负载的考虑因素
 - ▶ 精确描述用户的真实工作负载
- 执行和分析的考虑因素
 - ▶ 验证响应&找到系统瓶颈

1、创建一个性能测试 建立脚本

- 捕获用户和服务器的交互来创建测试
- 创建的测试结果在树视图 (tree view) 中显示为一系列访问过的页面
- 创建脚本时的关键考虑点
 - ▶ 消除编程的需要
 - ▶ 不同的用户, 不同的输入数据
 - ▶ 关联系统响应数据



2、创建一个性能测试 制定工作负载

- 测试被组装起来以执行
 - ▶ 进度表定义了脚本执行的顺序
 - ▶ 进度表定义了被用来生成负载的代理机器
 - Linux 和 Windows 代理

- 创建工作负载时的关键考虑：
 - ▶ 可以灵活的来精确建模复杂的工作负载
 - ▶ 在运行时可以动态的增加负载来找到临界点



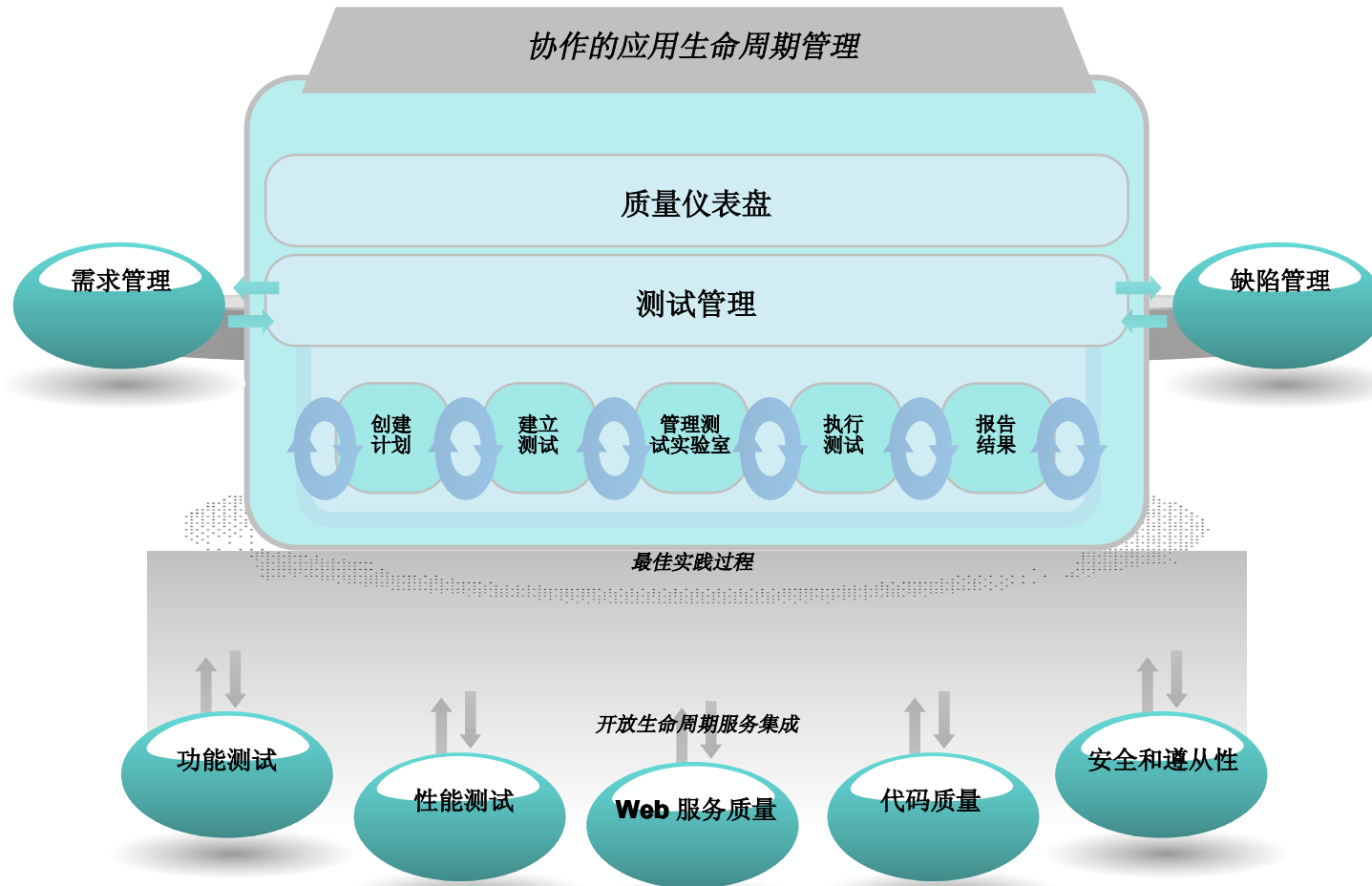
3、创建一个性能测试 执行和分析

- 中央控制台来执行分布式的测试
 - ▶ 主控制台可以协调和监控所有的活动
 - ▶ 可以用Windows和Linux的代理来生成负载

- 分析时的关键考虑
 - ▶ 找到响应速度慢的页面
 - ▶ 找到资源瓶颈



总结：质量架构



今天我们处于什么位置？ 我们期望处于什么位置？



