

# IBM Rational Quality Manager



# Rain Sensing Wiper: Example of a System Design Failure

## *Individual Systems Worked, But Failed When Integrated*

Windshield provided by local supplier

- ▶ Incompatible with the operation range of the sensor
- ▶ No captured requirement for proper system calibration (i.e., verifying sensor and windshield compatibility)
- ▶ *Cars were sent to customers with non-functioning wiper system*



Initial diagnostics designated software as culprit for malfunction

- ▶ Mechanics couldn't test software behavior
- ▶ Other components (electronic control unit, sensor, and windshield) functioned normally when tested independently
- ▶ *Failure was not of individual components, but in the interaction at a system level*



# The Cost of Poor Software Performance

## *A failed business-critical application can have a major impact on business*

**cnet news** [Search]

Home > News > Relevant Results

**Relevant Results** — By Tom Krazit

September 1, 2009 8:55 PM PDT

### Gmail outage blamed on capacity miscalculation

by Tom Krazit

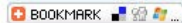
Font size | Print | E-mail | Share | 58 comments

Google's nearly two-hour Gmail outage Tuesday was the result of a miscalculation regarding the capacity of its system, the company said late Tuesday.

### eBusiness

## UK shoppers take a tough line on poor website performance

Author: [Antony Savvas](#)  
 Posted: 16:47 22 Sep 2008  
 Topics: [e-commerce](#) & [e-business](#)



UK online shoppers are increasingly unforgiving when they encounter [problems with retailers' websites](#).

Of the 90% of online shoppers who experienced problems in the past year, 49% abandoned their transaction, according to a survey into online consumer behaviour conducted by Harris Interactive for software firm Tealeaf.

**cnet news** [Search]

Home > News > Wireless

September 3, 2009 10:14 AM PDT

### Customers angered as iPhones overload AT&T

By Jenna Wortham  
 The New York Times

**nytimes.com**  
 The New York Times  
 2 comments

Slim and sleek as it is, the iPhone is really the Hummer of cell phones.

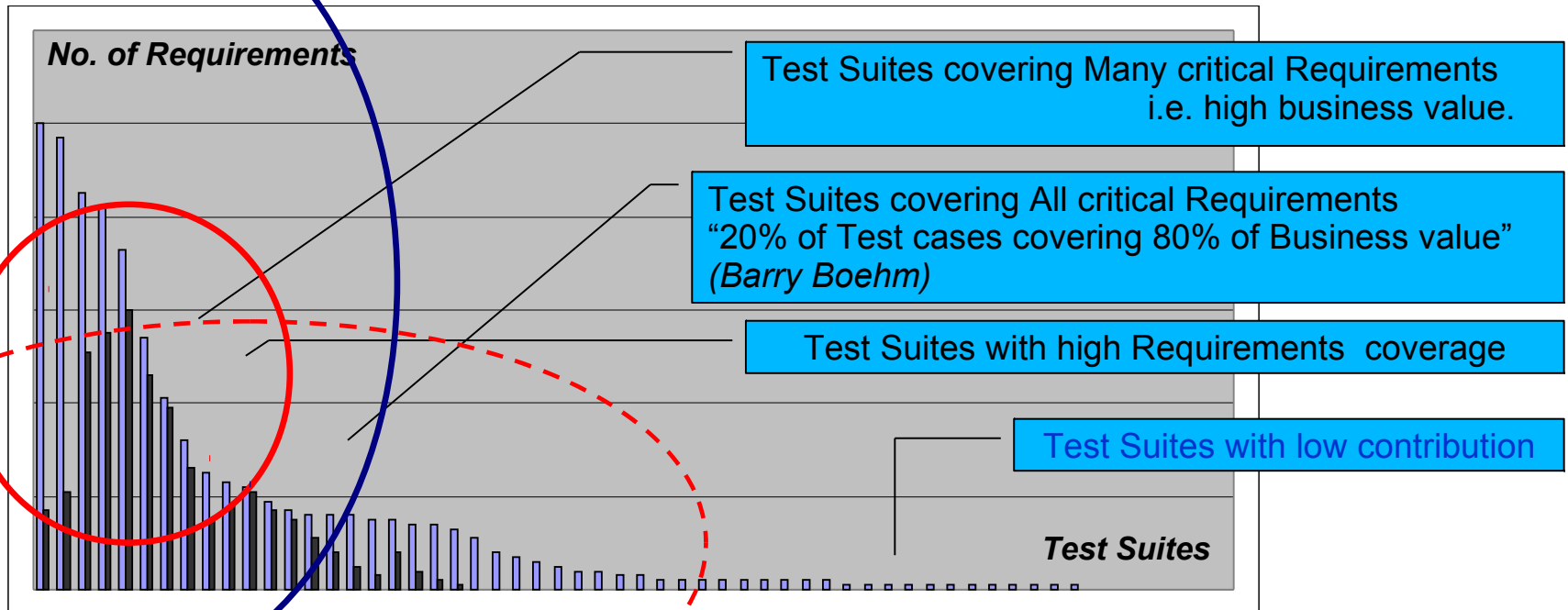
Owners use them like minicomputers, which they are, and not only do iPhone owners download applications, stream and browse the Web at higher rates than the average but the average iPhone owner can also use 10 times the data used by the average smartphone user.



# Applying automation where it matters the most

	No. of Changes
Module A	12
Module B	9
Module C	15
Module D	10
Module E	43
Module F	6

“80% of the defects come from 20% of the modules” (Barry Boehm)  
 => Most changed modules are often most defect-prone (Mockus, French)



# IBM Rational Quality Manager

## *A central hub for business-driven software quality*

### Mitigate business risk with collaboration

- ✓ Stakeholder and team coordination reduces mistakes
- ✓ Risk identification and management leads to educated prioritization decisions
- ✓ Test traceability linked to business requirements improves customer satisfaction

### Improve operational efficiency with automation

- ✓ Running tests earlier leads to reduced repair costs
- ✓ Running more tests in less time improves coverage
- ✓ Reducing manual labor leads to fewer testing errors
- ✓ Lab configuration automation improves efficiency and asset utilization

### Make confident decisions with effortless reporting

- ✓ Real-time dashboards enable proactive risk management
- ✓ Customizable reports facilitate ongoing process improvement



# Comprehensive dynamic planning and updates

## Process flow, not artifacts drives team activities

### Other Test Plans

Table of Contents

DOCUMENT HISTORY

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Overview

1.2 Test Objectives

1.3 Test Scope

1.4 Assumptions

1.5 Glossary

1.6 References

2 TEST STRATEGY

2.1 Business Functions

2.2 Structural Functions

2.3 Risk Assessment

2.4 Test Focus areas

2.5 Level of Testing

2.6 Functional and Structural Test Types

2.6.1 Test Focus Type matrix

2.6.2 Test Levels Type matrix

3 TEST PLAN

3.1 Roles and responsibilities

3.2 Test Schedule

3.3 Major testing subareas

3.4 Resource requirements

3.5 Testing for Non-Functional Requirements

4 TEST SUBJECT

4.1 Unattached

4.2 Application

4.3 Mercury Tours Application

4.4 Profiling

4.5 Flight Reservation

4.6 Select Flight

4.7 Select Flight Page

4.8 Book Flight

4.9 Flight Confirmation

4.10 Flight Cost

4.11 Flight Finder

4.12 Flight Reservation

4.13 Flight\_Reservation

4.14 Flight\_Reservation\_Stress

4.15 Cruise

4.16 Itinerary

4.17 Compiled Modules

4.18 Mercury Tours Site

Word based Test Plan

Test Plan Tree

### Rational Quality Manager Plan

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

+ Add New Section

- Live dynamic documentation
- Defines test process and strategy
- Defines responsibilities
- Activity based versus hierarchy
- Business level reporting against quality objectives



# Requirements driven testing

## *Knowing what to test*

View Requirements <sup>?</sup>

View Builder  
Show Requirements that match the attributes in the View Builder.

Group by: Ungrouped

10 Items per page Previous | 1 - 10 of 14 | Next

<input type="checkbox"/>	Status	ID	Risk	Name	Description	Owner
<input type="checkbox"/>		5		Data entry - change customer details	Confidential information for an existing account sha...	Coral Chen
<input type="checkbox"/>		2		Data entry - customer details	The system shall accurately capture basic custome...	Coral Chen
<input type="checkbox"/>	updated			Process mortgage increase - main path	The system shall process a valid mortgage increase...	Amber Alvarez
<input type="checkbox"/>		7		Forward mortgage to secondary approver	Ownership transfer of a mortgage increase request...	Dusty Dixon
<input type="checkbox"/>		9		View status of mortgage increase request	The system shall promptly and accurately display th...	Fern Farlow
<input type="checkbox"/>		6		Update mortgage application status	The system shall correctly update the status of a m...	Bridget Blue
<input type="checkbox"/>		4		Cancel an application	The system shall reliably cancel and archive a suspen...	Eliot Eggplant
<input type="checkbox"/>		15		Spelling accuracy and professionalism	Basic banking words like "amortization" shall be spelle...	Amber Alvarez
<input type="checkbox"/>		10		Display customer information	The system shall correctly display all customer acco...	Helen Hughes
<input type="checkbox"/>		13		Process mortgage request - nonexistent record	The system must reject an increase request that re...	Amber Alvarez

Previous | 1 - 10 of 14 | Next

- Requirements tracking built into the test management tooling
- Customizable attributes enable you to track what is important to your team
- Real-time impact analysis of requirements changes
- Traceability of test results to user needs

Know you are testing the right things



# Collaborative risk based testing

## *Risk management and prioritization*

**Classics Java Test Plan**  
Test Plan Overview | View Snapshots

Originator: ADMIN Action: Select Action

Test coverage for the Classics Java application

**Test Cases**

Lists the test cases associated with a given plan. Removing a test case will remove the association.

Group by: Ungrouped

Show All Items per page

ID	Risk Assessment	Suspect	...
12	○○○○○	◇	vsme
15	○○○○○	◇	A
16	○○○○○	◇	C
17	○○○○○	◇	A
18	○○○○○	◇	A
19	○○○○○	◇	C

**Risk Assessment:**  
○○○○○ very high

**My Risk:**  
Rate this: ○○○○○ very low  
Comment here

**Community Risk:**

Very high: 0 (0%)  
High: 0 (0%)  
Neutral: 0 (0%)  
Low: 0 (0%)  
Very low: 1 (100%)  
average ○○○○○ very low

**Work Item:** Create

Type Filter Text

action	Theme	Weight	Modified
ction	Functiona...	20	1 minu...
ction	Functiona...	100	1 minu...
ction	Functiona...	100	1 minu...
ction	Functiona...	100	1 minu...
ction	Functiona...	100	1 minu...
ction	Functiona...	100	1 minu...

- Risk assessments captured in Test Plan and Test Cases
- Collaboration planning of risk mitigation strategy

- Test Case will contain a risk failure score and a risk priority score
- Documented risk related decisions

Base project decisions on qualitative risk analysis



# Test coverage optimization

*Focus resources on testing the right combinations*

## Test Configurations

- 4 languages
- 4 browser types
- 5 databases
- 5 application servers
- 400 Combinations!



## Pairwise Optimizations

## Test the right 20 combinations

OS	Browser	Protocol	CPU	DBMS
XP	IE	IPv4	Intel	MySQL
XP	Firefox	IPv6	AMD	Sybase
XP	IE	IPv6	Intel	Oracle
OS X	Firefox	IPv4	AMD	MySQL
OS X	IE	IPv4	Intel	Sybase
OS X	Firefox	IPv4	Intel	Oracle
RHL	IE	IPv6	AMD	MySQL
RHL	Firefox	IPv4	Intel	Sybase
RHL	Firefox	IPv4	AMD	Oracle
OS X	Firefox	IPv6	AMD	Oracle

## Configuration awareness

### Test Platform Environment Management

- Focus your environment coverage
- Document your environment coverage
- Gain agreement across the project



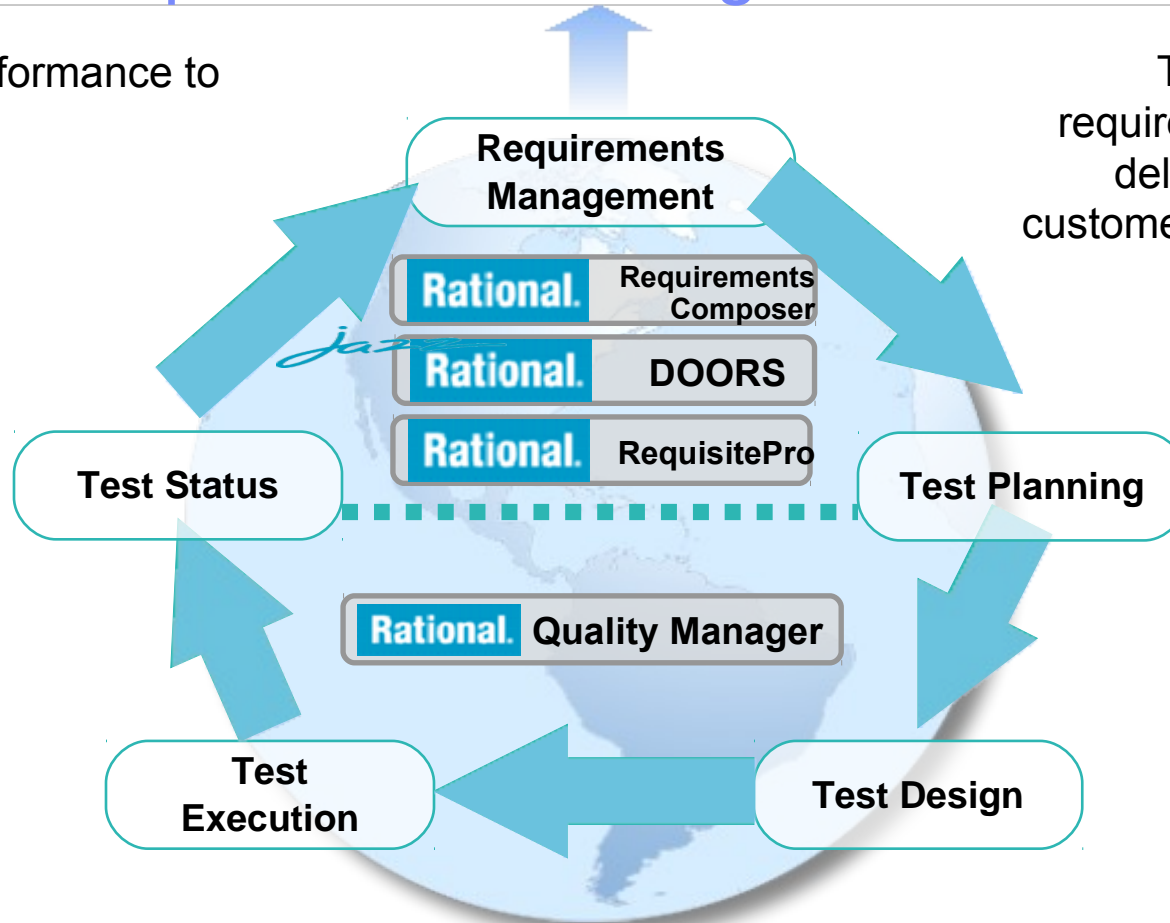
**Test the right cases instead of everything. Plan optimal execution**



# Integrated Requirements Management

Quality is conformance to requirements

Tests based on requirements ensure deliverables meet customer expectations



**Process Automation and Increased Focus**  
The test team is working against the right set of requirements

# Integrated Manual test authoring and execution

Step	Description	Expected Results
1	Start Classics Application	This window should appear.
2	Select a CD. User browses to Schubert and selects String Quarter	
3	Order the selected CD. Press the Place Order Button. The Place Order Window appears with Schubert as a selected item.	
4	Insert CC Number	
5	Validate Trent Cuplit	
6	Press the	
7	Press the OK Button	
8	OK Button	

#	Type	Result	Description	Attachment
1	Step	Passed	Select a cd and click Order button	
2	Step	Passed	Verify the login window displays	
3	Step	Passed	Select new customer radio button and select OK button	

- Manual test author and execute
  - ▶ Step by step capture and execution of manual tests
  - ▶ Assisted data entry
  - ▶ Keyword support for integrated manual and automated testing
  - ▶ Rich defect capture during execution, including screenshot and attachments
  - ▶ Simple intuitive interface for quick test execution

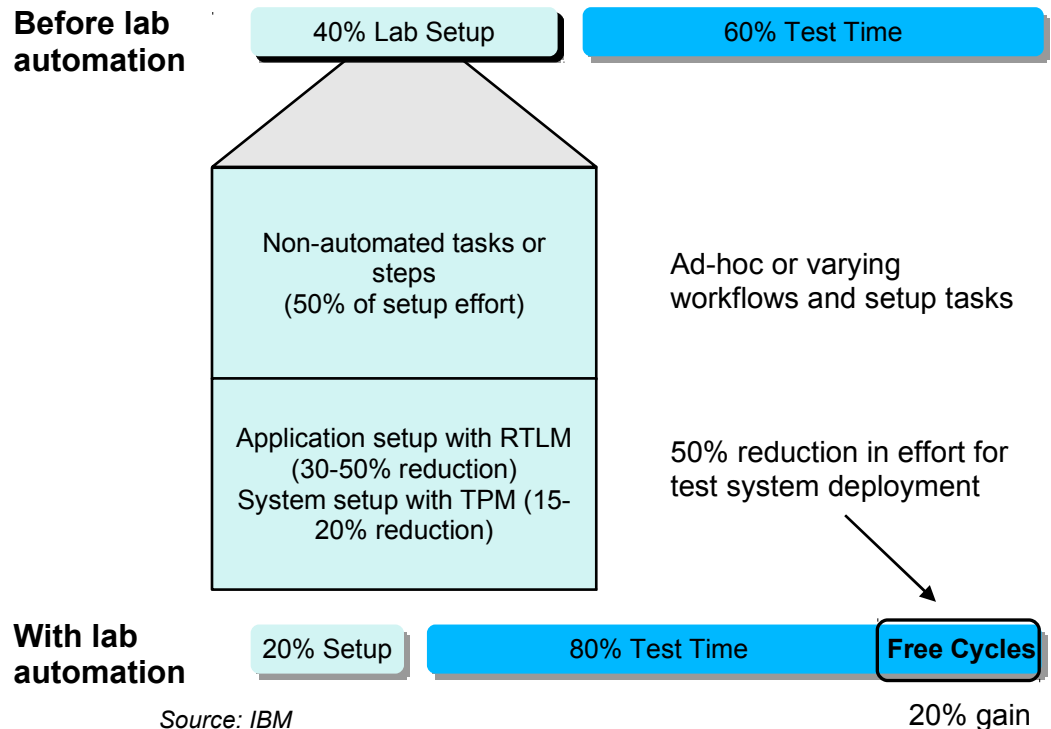
Maximizing efficiency of manual testing



# Test lab automation and management

## *Insight and control over the test lab*

- Manage
  - ▶ Verify that I have the resources required to fulfill my test plan
- Deploy
  - ▶ Deliver the configurations my teams require for test
- Optimize
  - ▶ Analyze patterns to minimize cost and maximize utilization

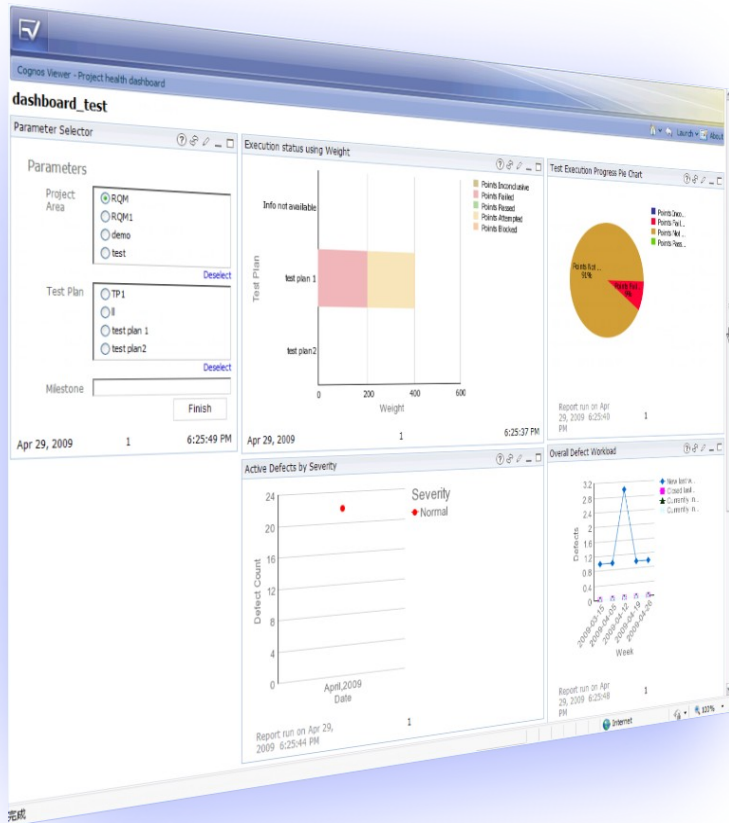


**Work smarter, save on test lab overhead, infrastructure and duration costs**



# Make confident decisions with effortless reporting

## *Closed Loop Analysis & Reporting*



- Customizable reports and dashboards
  - ▶ Reduce escalating cost of information gathering
  - ▶ Reduce risk by identifying trends before they become issues
  - ▶ Raise enterprise visibility and transparency to reduce costs and risk
  - ▶ Measures the effectiveness of processes and practices to improve organizational and business outcomes

Make the right decisions at the right time



# IBM Rational Functional Tester



# Test automation has it's own challenges

*How to effectively implement test automation while minimizing overhead*



**Cost:** Realize fast time to value and low long-term cost of maintenance.



**Complexity:** Assure that all software in the organization is tested effectively, independent of platform, application or environment.



**Responsiveness:** Enable the test team to efficiently create and execute tests while maximizing the ability to reuse and enhance existing scripts.

# Maximize your investment in test automation

## *With IBM Rational Functional Tester*

- Achieve success quickly and minimize maintenance
  - ▶ Simplified natural language scripting with Storyboard testing
  - ▶ Eclipse based or Visual Studio .net
  - ▶ Easy to learn
  - ▶ Maximize reuse
- Complete test coverage
  - ▶ Supports testing for Java, Web, Visual Basic .Net, SAP, Siebel, Web 2.0, Power Builder and Terminal Based applications
  - ▶ Ability to support custom controls

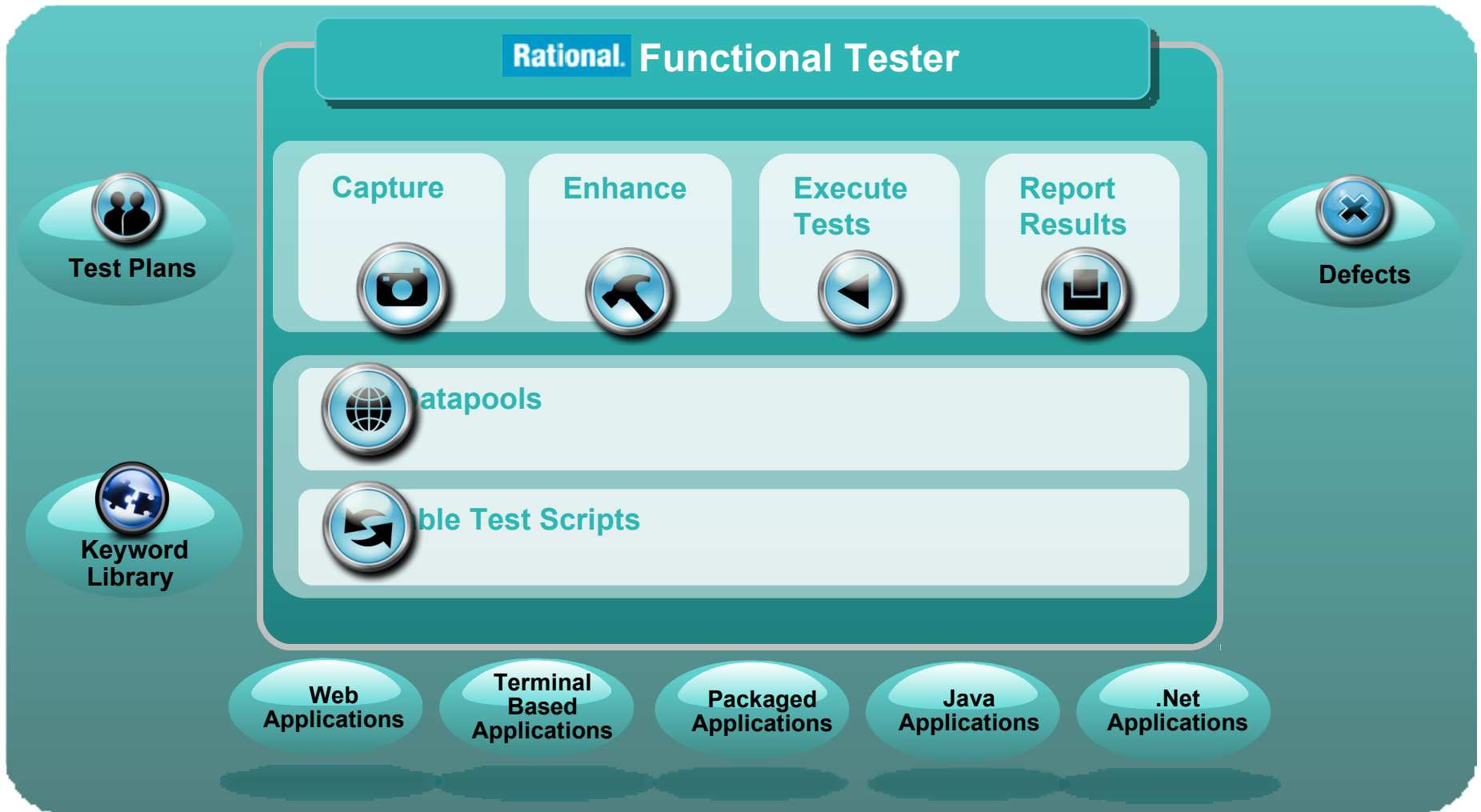


System z

System i



# Rational Functional Tester 8.1



# Demo: A team collaborates across repositories

**Bob – Product Owner**



- Defines & links requirements to influence the development plan

*Plan Requirements*

**Scott, Deidre, Marco – Developers**



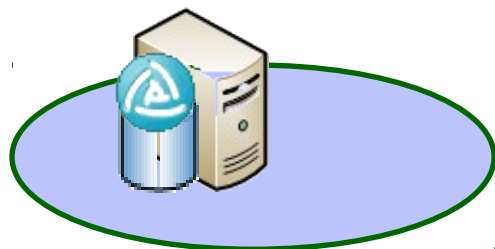
- Plan the iteration from requirements
- Develop & deliver changes

*Find & Fix Defects*

**Tanuj – Tester**



- Validate requirements
- Submit defects
- Test the iteration



Rational Requirements Composer



Rational Team Concert

*Requirements driven testing*



Rational Quality Manager

