

# Innovate2010

The Rational Software Conference

Let's **build** a smarter planet.

**Ritesh Prasad**

**[riprasad@in.ibm.com](mailto:riprasad@in.ibm.com)**

Integrating your  
Collaborative  
Development, Build and  
Test Environments

The premier software and product delivery event.



## Agenda

### **Business Problem**

### **Challenges with Effectively Managing Development**

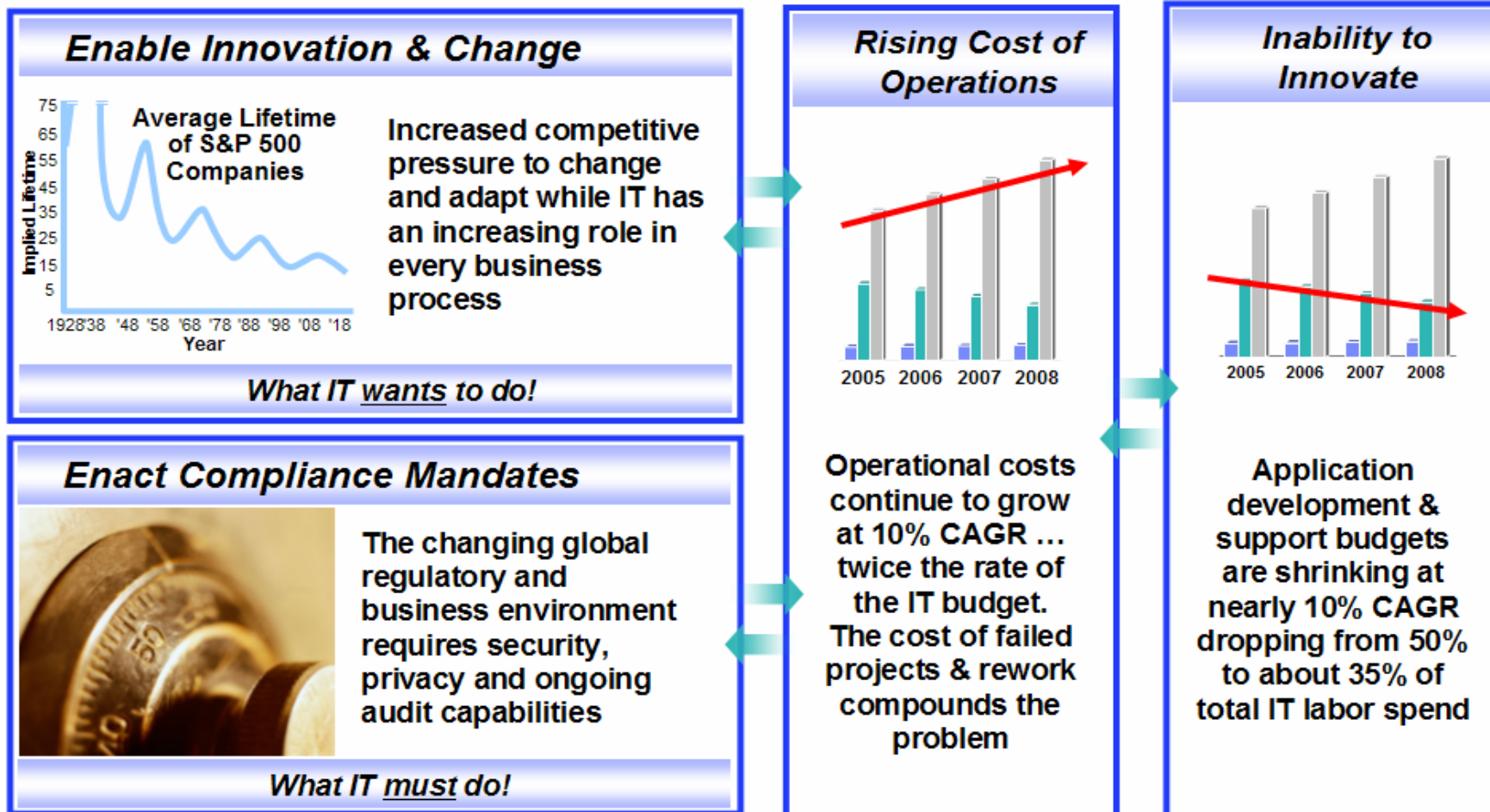
### **Integration Scenarios**

Feedback metrics help identify shortfalls

Collaborative problem determination

## The landscape facing IT & business leaders

The need to innovate with less



**Bottom Line: Enable Business Flexibility...**

## Software development and delivery challenges in times of innovation and constant change



*"Only 70% of software projects are successful, costing over \$300B annually"*



*"Only 37% are satisfied with the speed of software development"*



*"50% of outsourced projects are expected to under perform"*



*"Only 42% of users are satisfied with quality"*

***Caused by Silos of people, process, and projects***

### Geographic Barriers

- Poor communication
- Language, culture, time
- Process gaps resulting in rework
- High degree of friction

### Organizational Barriers

- Lack of meaningful collaboration
- Weak project governance
- Lack of domain expertise
- Poor LOB oversight
- Security of IP when outsourcing

### Infrastructure Barriers

- Incompatible tools / repositories
- Unreliable access artifacts
- Lengthy on-boarding
- Inflexible tooling integration

## Agenda

### **Business Problem**

## **Challenges with Effectively Managing Development**

### **Integration Scenarios**

Feedback metrics help identify shortfalls

Collaborative problem determination



## Challenges with Effectively Managing Development

*We need a proven and stable set of tools that can handle our entire development process*

*We need to be able to track and deliver project status to all stakeholders*



Deploy

*How do we minimize the challenges moving the application from development to production?*

Develop

*How can I quickly deliver working software that scales and integrates easily?*

Test, Analyze

*How can we enable teams to build in quality and security, and ensure we are validating business requirements along the way?*

Reuse

*How does development best manage their work products? How do we enable reuse across different projects?*

Build

*How do we get more visibility and accountability in the builds. Are the builds stable, did the needed fixes get in...?*

**Collaborative  
IT Development  
For  
IBM middleware**

Teamwork

*How do I collaborate with geographically diverse teams to ensure clear communication of joint goals?*



## Challenges with Effectively Managing Development

*We need a proven and stable set of tools that can handle our entire development process*

*We need to be able to track and deliver project status to all stakeholders*



Deploy

*How do we minimize the challenges moving the application from development to production?*

Develop

*How can I quickly deliver working software that scales and integrates easily?*

Test, Analyze

*How can we enable teams to build in quality and security, and ensure we are validating business requirements along the way?*

Reuse

*How does development best manage their work products? How do we enable reuse across different projects?*

Build

*How do we get more visibility and accountability in the builds. Are the builds stable, did the needed fixes get in...?*

Teamwork

*How do I collaborate with geographically diverse teams to ensure clear communication of joint goals?*



## RAD: Tools for all tiers of IT application development

### Development Tools:

#### 1. Productive tools for initial creation and updates of artifacts (web pages, EJBs, Web services, etc).

- Editors (source and graphical), wizards, validators, refactoring and quickfixes

#### 2. Database query optimization

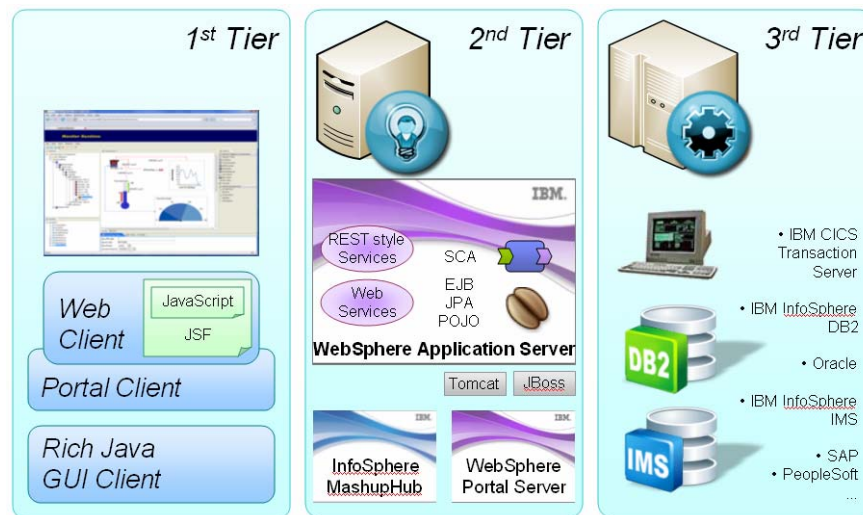
- Generate java data access layers
- Identify application hot spots
- Enhance problem isolation
- Analyze impact of changes and use of sensitive data
- Use JDBC and SQL data access best practices
- Use DB2 static SQL for performance and manageability
- Replace problem SQL without opening the application

#### 3. Dynamic testing, debugging, & profiling

- Deploying to WAS UTE, Universal Test Client to test Java or EJBs, Web Services Explorer, debuggers for each domain
  - eg. Java, XSLT, Java Script, etc., Profiler

#### 4. Migration & N-2 support – from previous versions of RAD, also from previous spec levels.

Target 2 previous server versions, for application creation and deployment (testing).





## Challenges with Effectively Managing Development

*We need a proven and stable set of tools that can handle our entire development process*

*We need to be able to track and deliver project status to all stakeholders*



Deploy

*How do we minimize the challenges moving the application from development to production?*

Develop

*How can I quickly deliver working software that scales and integrates easily?*

Test, Analyze

*How can we enable teams to build in quality and security, and ensure we are validating business requirements along the way?*

Reuse

*How does development best manage their work products? How do we enable reuse across different projects?*

Build

*How do we get more visibility and accountability in the builds. Are the builds stable, did the needed fixes get in...?*

Teamwork

*How do I collaborate with geographically diverse teams to ensure clear communication of joint goals?*

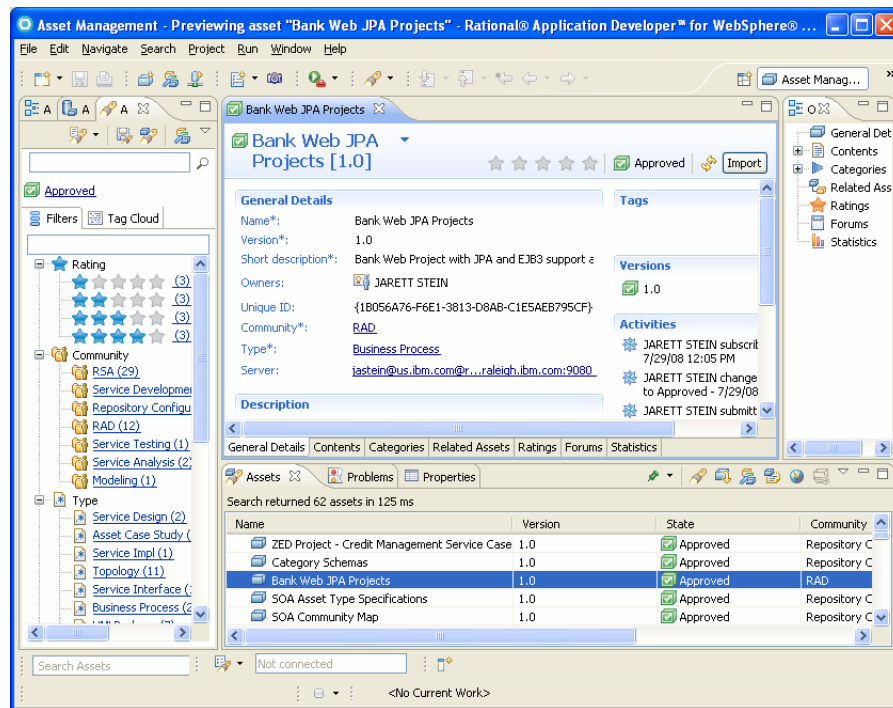
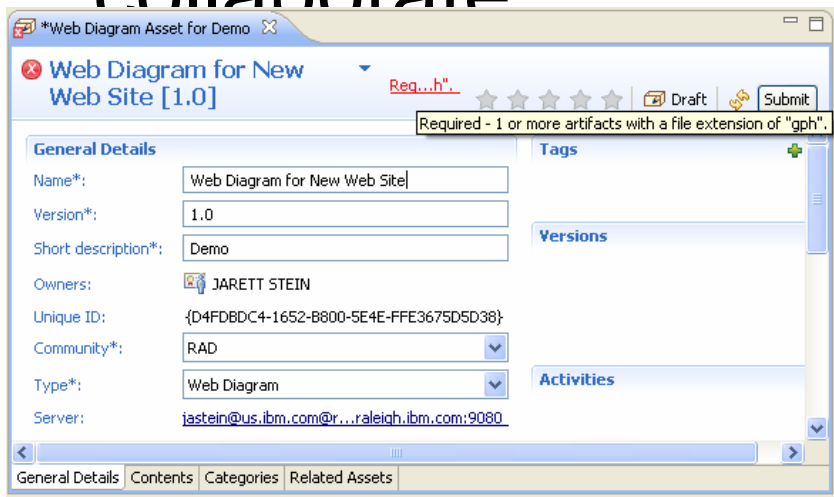


**Collaborative  
IT Development  
for  
IBM middleware**

## IBM Rational Asset Manager

Search, import, and submit reusable artifacts

Submit for review and collaborate



## Challenges with Effectively Managing Development

*We need a proven and stable set of tools that can handle our entire development process*

*We need to be able to track and deliver project status to all stakeholders*



**Deploy**

*How do we minimize the challenges moving the application from development to production?*

**Develop**

*How can I quickly deliver working software that scales and integrates easily?*

**Test, Analyze**

*How can we enable teams to build in quality and security, and ensure we are validating business requirements along the way?*

**Reuse**

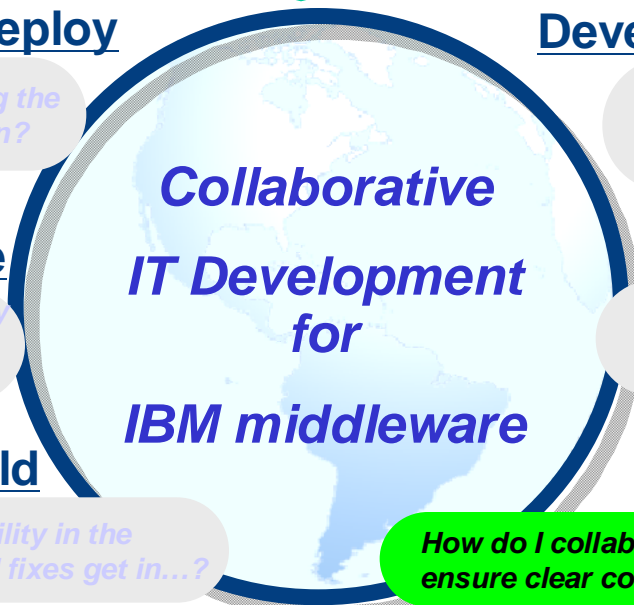
*How does development best manage their work products? How do we enable reuse across different projects?*

**Build**

*How do we get more visibility and accountability in the builds. Are the builds stable, did the needed fixes get in...?*

**Teamwork**

*How do I collaborate with geographically diverse teams to ensure clear communication of joint goals?*



**Collaborative  
IT Development  
for  
IBM middleware**

## IBM Rational Team Concert

*Software innovation through collaboration*

- Enables “real-time, in-context” collaboration for distributed project teams – making software development more automated, transparent and predictive
- Integrates source control, work item, reporting and build capabilities which “think and work in unison”
- Provides real-time project health information and transparency of status through automated data gathering
- Supports out-of-the-box and custom defined processes which break down functional & organizational barriers
- Allows choice of client tools and extends the value of ClearQuest & ClearCase in enterprise deployments

Open and extensible on

- ✓ Collaborate in context
- ✓ Right-size governance
- ✓ Day one productivity

### IBM Rational Team Concert



transparent *integrated presence*  
wikis OPEN real-time reporting  
chat automated hand-offs Web 2.0  
*custom dashboards* automated data gathering  
**EXTENSIBILITY** Eclipse plug-ins services  
architecture **FREEDOM TO CREATE**

JAZZ TEAM SERVER

## In-context Collaboration

### Team Awareness

Shows team members and their online status  
Shows what they are working on

### Team Central

News & events  
Build status  
What's being worked on  
Changes

RSS Feeds  
Configurable  
Personalizable

The screenshot displays the IBM Rational software interface with several key components:

- Team Awareness:** A sidebar on the left shows team members with their online status (e.g., 'No Work Time Left', 'Hours: 0/0', 'Estimated: 0%').
- Team Central:** A central pane displays news and events, including items like 'Gorkem Ercan: Eclipse moment of competition' and 'Nick Boldt: Managing Plugins and Features'. Below this is a 'Team Load' section with a bar chart showing work items by priority (Unassigned, Low, Medium, High).
- Collaboration:** A chat window is open, showing a conversation between Kevin Haaland and kartik@ca... regarding a story (45274).
- Work Items:** A 'My Open Work Items' section shows a bar chart of work items by severity (Unassigned, Minor, Normal, Major, Critical, Blocker).
- Event Log:** A section on the right shows a list of events, including 'Outgoing synchronization error (WorkItem)' and 'Improvements to QCert reports (46379)'.

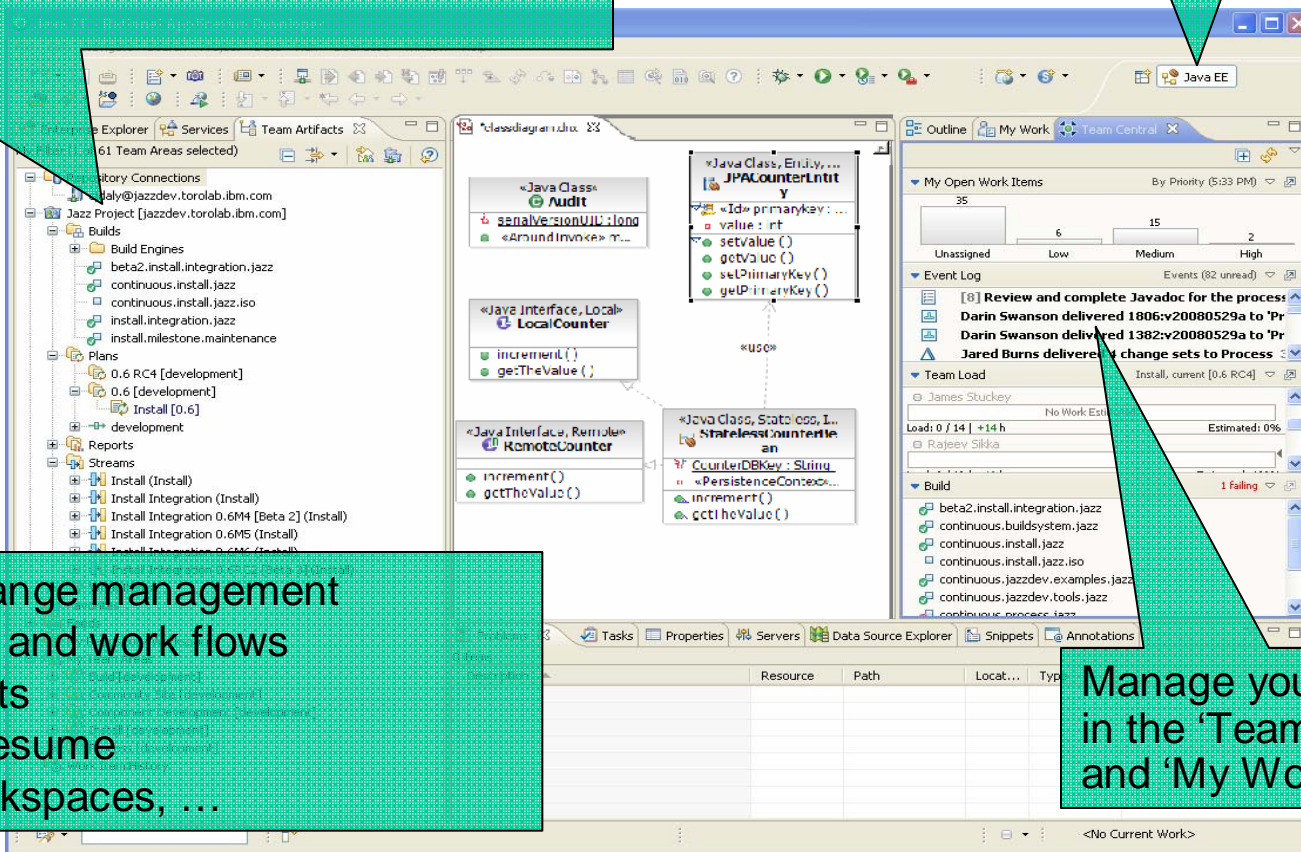
Collaborate in context

## RTC Views & Capabilities integrate with RAD & RSA

Work with your Web & Java EE development artifacts in the 'Team Artifacts' view

- code
- diagrams
- Metadata
- ...

Develop in RAD and RSA perspectives (Modeling, Java EE, ...)



Use Jazz change management constructs and work flows

- change sets
- suspend/resume
- server workspaces, ...

Manage your workload in the 'Team Central' and 'My Work' views

## Challenges with Effectively Managing Development

*We need a proven and stable set of tools that can handle our entire development process*

*We need to be able to track and deliver project status to all stakeholders*



**Deploy**

*How do we minimize the challenges moving the application from development to production?*

**Develop**

*How can I quickly deliver working software that scales and integrates easily?*

**Test, Analyze**

*How can we enable teams to build in quality and security, and ensure we are validating business requirements along the way?*

**Reuse**

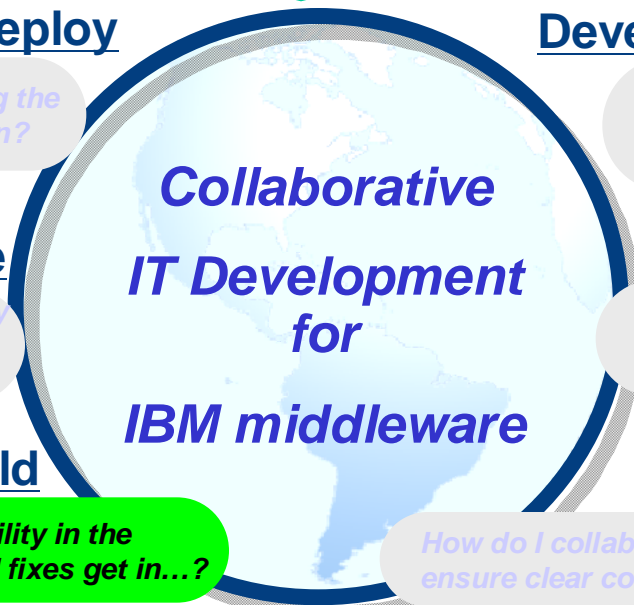
*How does development best manage their work products? How do we enable reuse across different projects?*

**Build**

*How do we get more visibility and accountability in the builds. Are the builds stable, did the needed fixes get in...?*

**Teamwork**

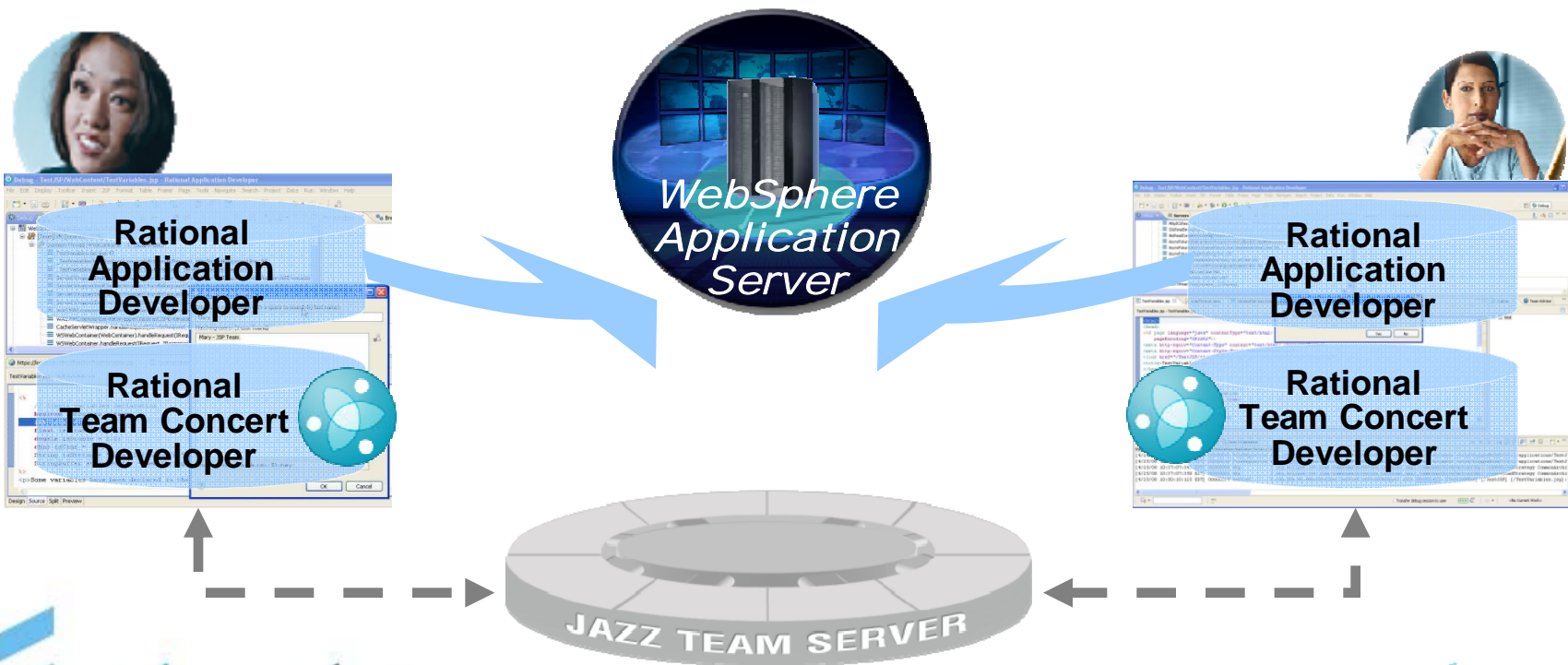
*How do I collaborate with geographically diverse teams to ensure clear communication of joint goals?*



## Extend team collaboration to IBM middleware developers

RAD and RTC integrate to:

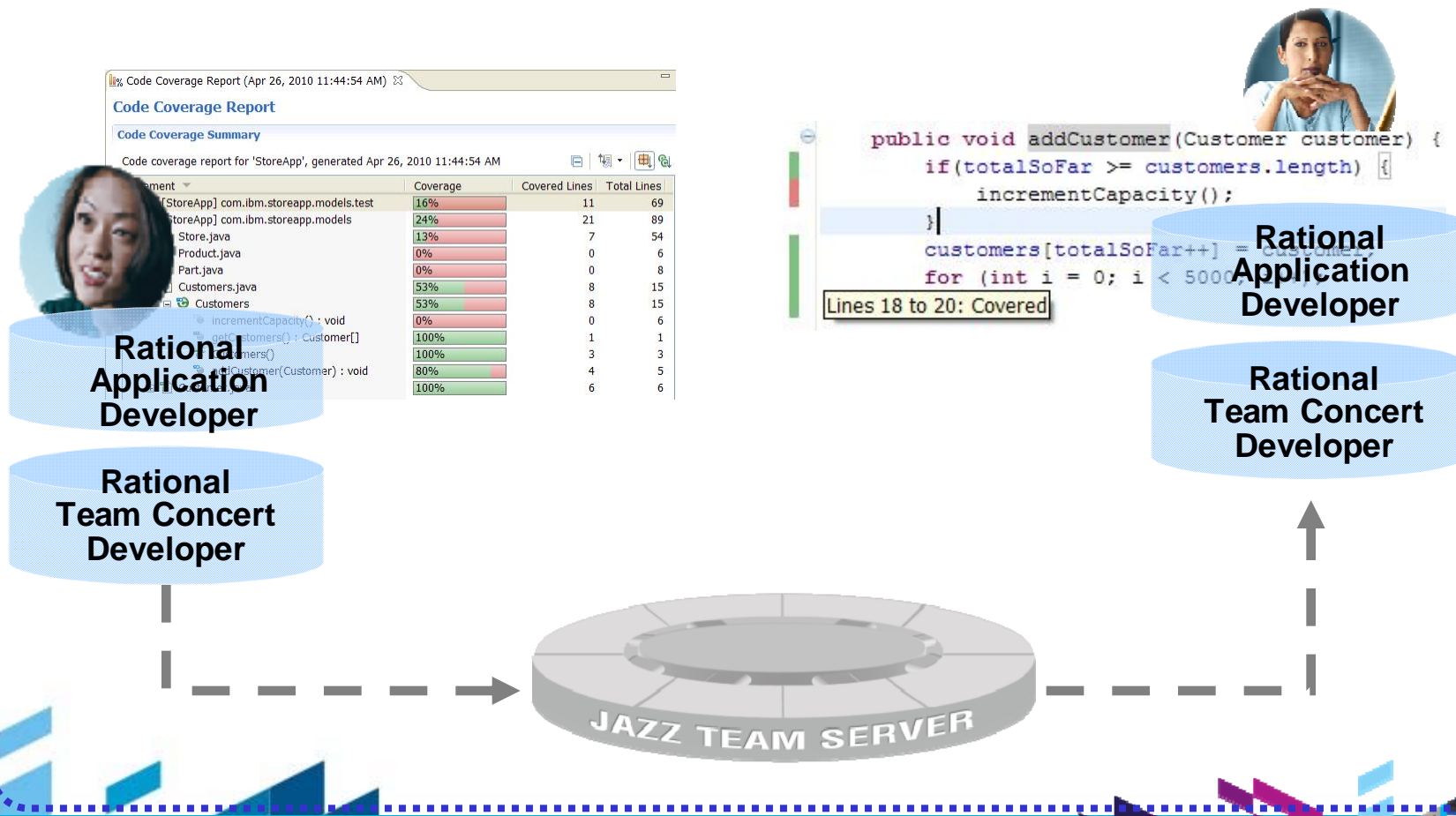
- ⊕ Manage SCM, Build, and work items from *within* RAD
- ⊕ Share live debug sessions between team members





Extend team collaboration to IBM middleware developers  
- deliver better tested software early in the cycle (RAD *Next Beta*)

- Share code coverage information from automated testcase execution
- Improve test coverage and quality based on code coverage results



## Challenges with Effectively Managing Development

*We need a proven and stable set of tools that can handle our entire development process*

*We need to be able to track and deliver project status to all stakeholders*



Deploy

*How do we minimize the challenges moving the application from development to production?*

Develop

*How can I quickly deliver working software that scales and integrates easily?*

Test, Analyze

*How can we enable teams to build in quality and security, and ensure we are validating business requirements along the way?*

Reuse

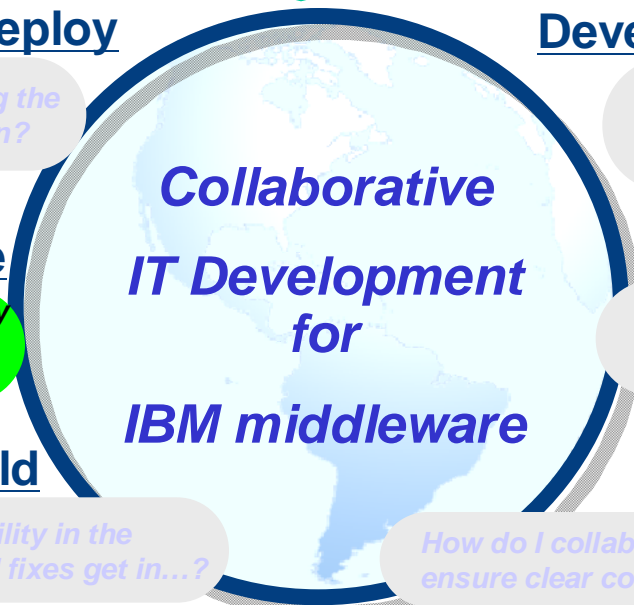
*How does development best manage their work products? How do we enable reuse across different projects?*

Build

*How do we get more visibility and accountability in the builds. Are the builds stable, did the needed fixes get in...?*

Teamwork

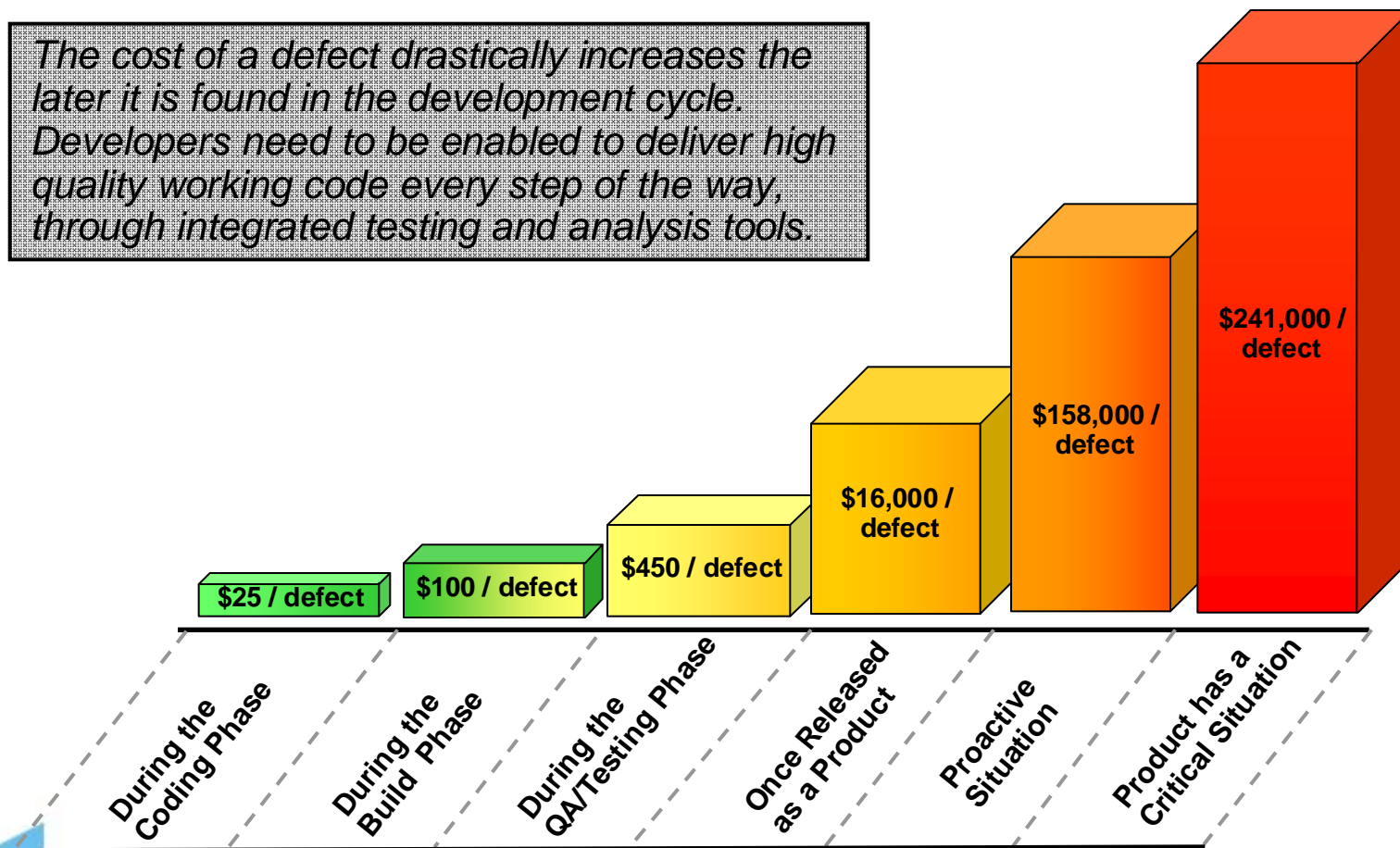
*How do I collaborate with geographically diverse teams to ensure clear communication of joint goals?*



**Collaborative  
IT Development  
for  
IBM middleware**

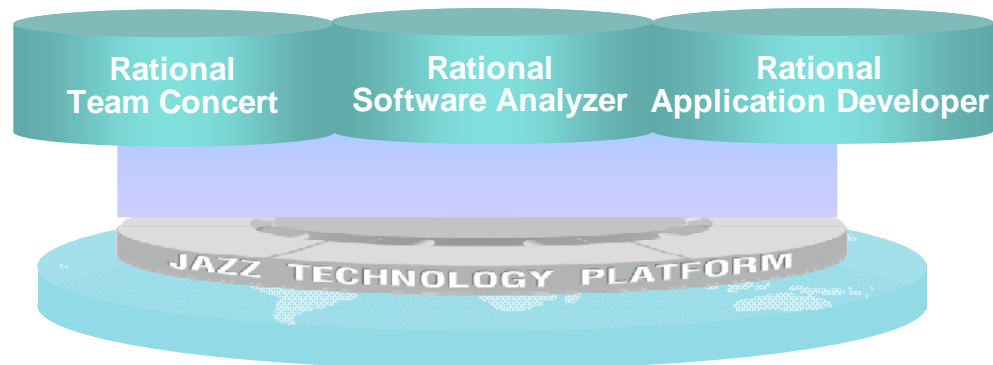
## What is the Cost of Quality?

*The cost of a defect drastically increases the later it is found in the development cycle. Developers need to be enabled to deliver high quality working code every step of the way, through integrated testing and analysis tools.*



## Rational Software Analyzer

Optimizing Development through effortless code quality Governance



Implement Code Quality Governance directly into the development stream

Developers can run configured rule sets before checking into repository

Measure team performance through visibility into globally dispersed development teams against best practice metrics

Manage adherence through quality reporting views in RSAR

## Challenges with Effectively Managing Development

*We need a proven and stable set of tools that can handle our entire development process*

*We need to be able to track and deliver project status to all stakeholders*



Deploy

Develop

*How do we minimize the challenges moving the application from development to production?*

*How can I quickly deliver working software that scales and integrates easily?*

Test, Analyze

Reuse

*How can we enable teams to build in quality and security, and ensure we are validating business requirements along the way?*

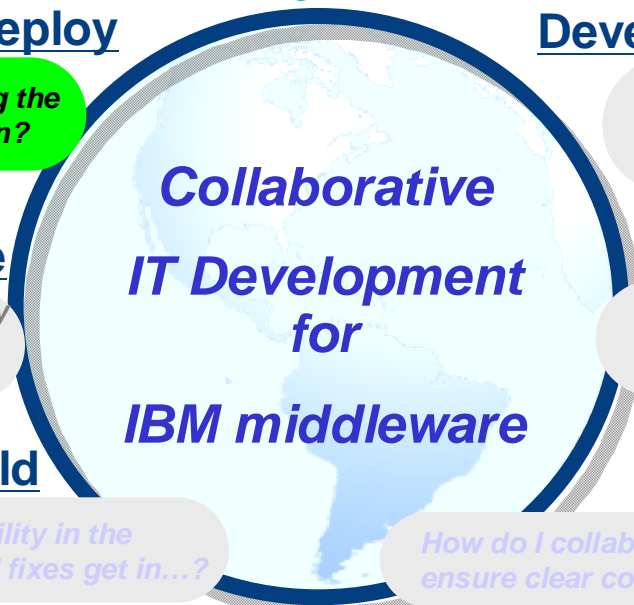
*How does development best manage their work products? How do we enable reuse across different projects?*

Build

Teamwork

*How do we get more visibility and accountability in the builds. Are the builds stable, did the needed fixes get in...?*

*How do I collaborate with geographically diverse teams to ensure clear communication of joint goals?*



## Goals for Application Deployment

### Reduce Maintenance Time

- Minimize deployment/environment build time
- Build Environments in hours not days
- Deploy code in minutes not hours

### Gaining Control of your Environment

- Eliminate Configuration drift problem
- Enforce Standards Across the Enterprise
- Put Configuration under Change Control



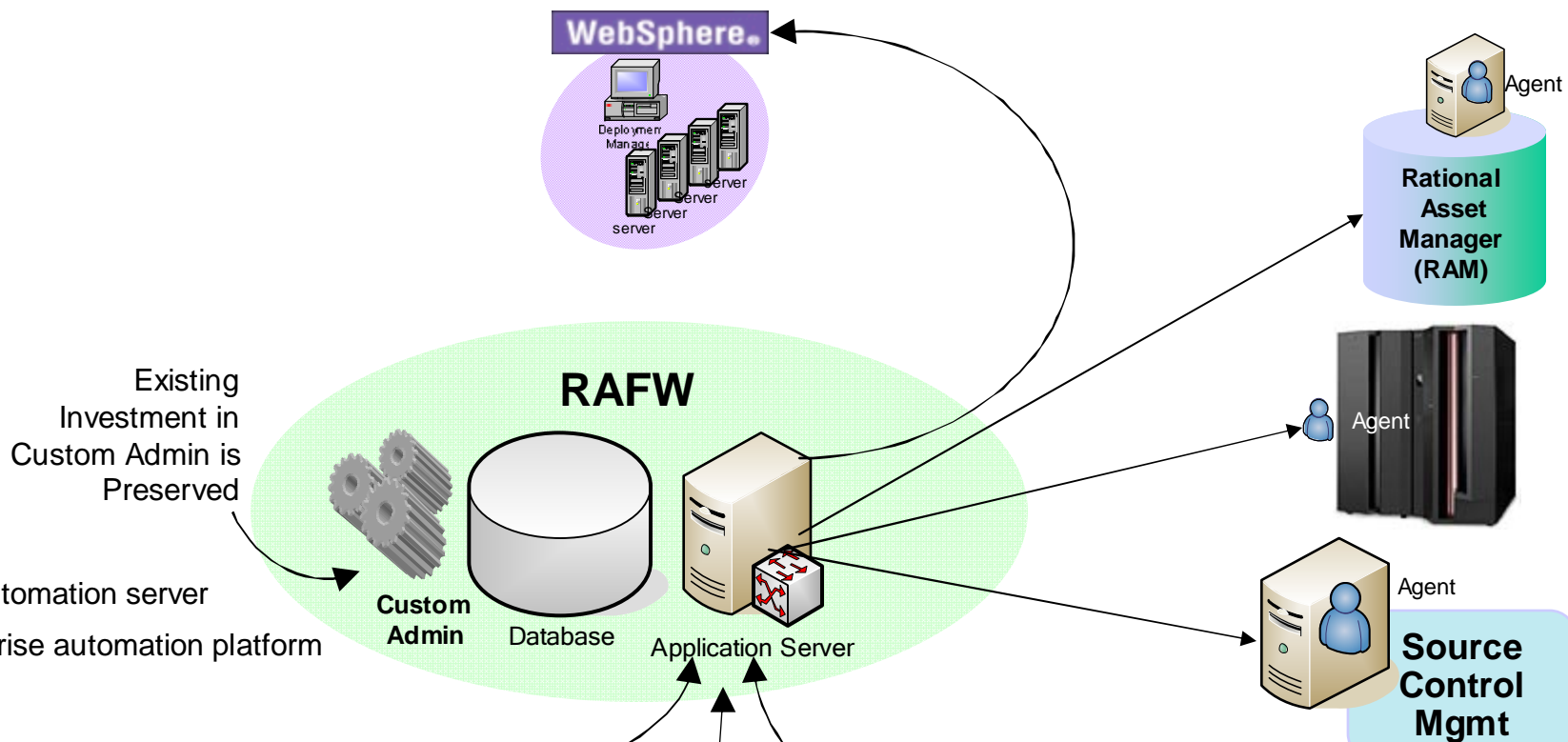
### Repeatability & Consistency

- Rebuild environments consistently
- Reliable and Repeatable Disaster Recovery
- Eliminate Human Errors

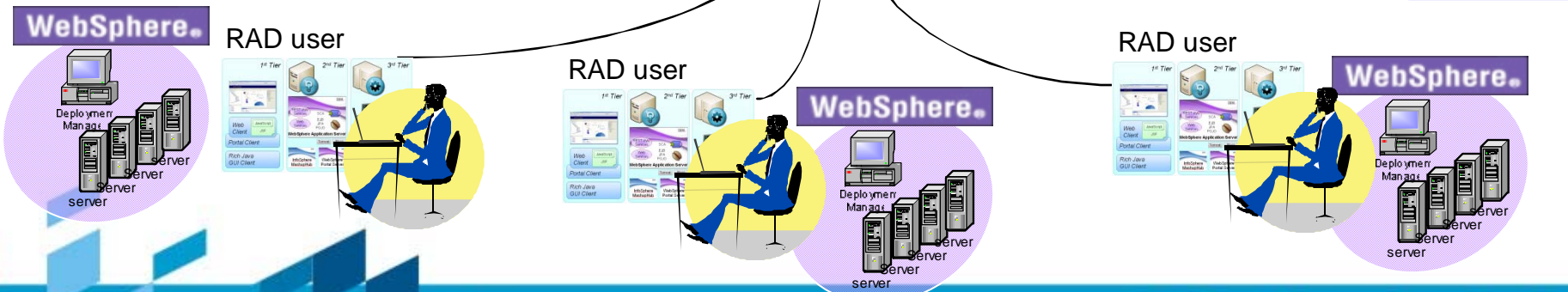
### Reduce Costs

- Free up staff for strategic initiatives
- Reduce labor costs
- If Time = Money, Less time = More Money

## RAFW - robust server configuration management



- Centralized automation server
- Proven enterprise automation platform



## Agenda

### **Business Problem**

### **Challenges with Effectively Managing Development**

#### **Integration Scenarios**

Feedback metrics help identify shortfalls

Collaborative problem determination





## Scenario #1 – Feedback metrics help identify shortfalls

### Process feedback

*The project lead will see that the code coverage results are at expected levels and resolves the defect*

### Process feedback

*Project lead notices a drop in code coverage from previous build*

### Work item resolved

*Developer checks code into a build and resolves defect when he sees the code coverage results for the build*

### Development

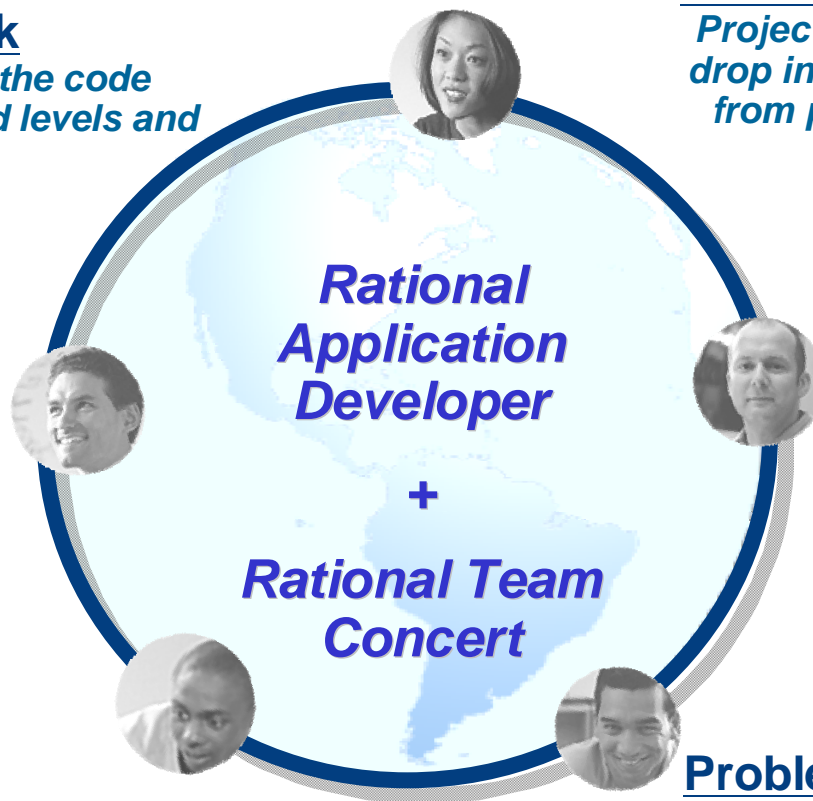
*A defect is opened against the build that had the noticeable code coverage level drop*

### New tests added

*New tests are added to cover the code that has not been covered by previous tests. Runs tests locally to ensure improved coverage*

### Problem determination

*Developer reviews build results and imports the code coverage results directly into the source code to see what source is missed during the test runs*



## Code Coverage

- Provides detailed information on what code paths have been encountered during program execution
- Powerful tool to help determine xUnit test coverage, potential dead code
- Command line and Ant capability for build integration
  - JUnit, code coverage data collection and html report generation

The screenshot displays an IDE interface with two main panes. The left pane shows a project tree for 'StoreApp' with the following structure and coverage percentages:

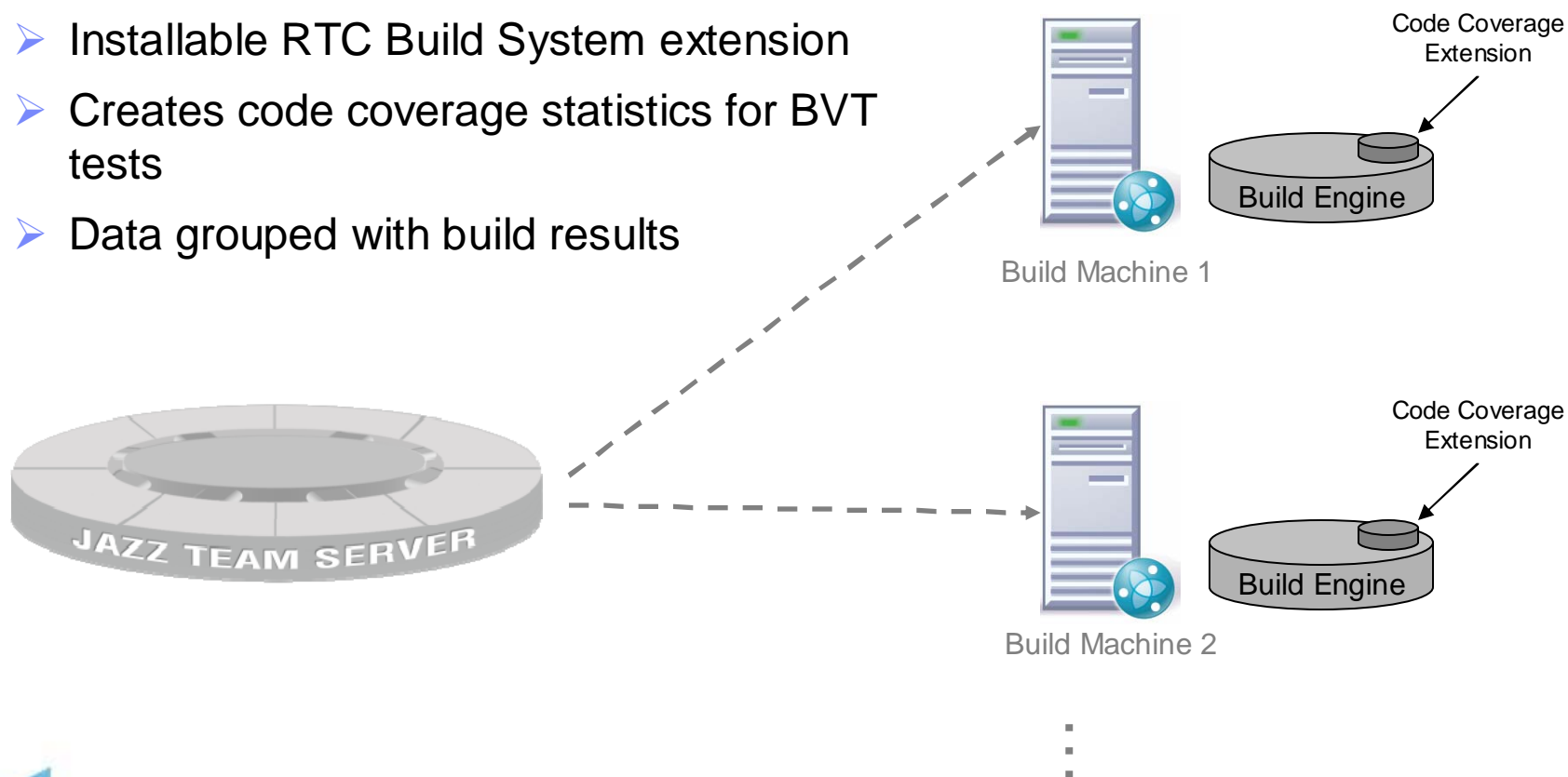
- StoreApp
  - com.ibm.storeapp.models [23%]
    - Customer.java [100%]
    - Customers.java [50%]
    - Part.java [0%]
    - Product.java [0%]
    - Store.java [13%]
  - com.ibm.storeapp.models.test [17%]
  - JRE System Library [jdk]
  - JUnit 3
  - Referenced Libraries
  - Class Diagrams

The right pane shows the source code for 'Customers.java' with a vertical bar on the left indicating coverage. The code is as follows:

```
private Customer[] customers;  
private int totalSoFar;  
  
public Customers() {  
    customers = new Customer[INITIAL_CAPACITY];  
}  
  
public void addCustomer(Customer customer) {  
    if(totalSoFar >= customers.length) {  
        incrementCapacity();  
    }  
    customers[totalSoFar++] = customer;  
}  
  
private void incrementCapacity() {  
    int currentCapacity = customers.length;  
    int doubleCapacity = currentCapacity * 2;
```

## Code Coverage Integration with RTC (RAD Next Beta)

- Build system integration
  - Installable RTC Build System extension
  - Creates code coverage statistics for BVT tests
  - Data grouped with build results



## Code Coverage Integration with RTC (RAD Next Beta)

### Client side

- RTC Build details viewer extension – RAD and Web browser
  - Show summary of code coverage statistics
  - Additional Code Coverage tab to show detailed coverage statistics report
- Integrated work item search and creation
- Import to local workspace for rich viewing in navigator and source views

Build StoreApp Project build w/ Code Coverage 20100421-1319

✓ **Completed**

Duration: 1 minute, 2 seconds  
 Start Time: April 21, 2010 4:19:44 PM  
 Completed: April 21, 2010 4:20:47 PM

Status Trend:

**Coverage:** type: 40%, method: 32%, block: 19%, line: 20%

JUnit: 1 test, 0 failures, 0 errors  
 Logs: 2 logs

Repository Workspace: Build Stream Workspace  
 Snapshot: StoreApp Project build w/ Code Coverage\_20100421-1319

Work items: None included  
 Coverage: type: 40%, method: 32%, block: 19%, line: 20%

Associated Release  
 Released builds are available as choices in the work item "Found In" field.  
[Create a release to associate with this build](#)

**Coverage**

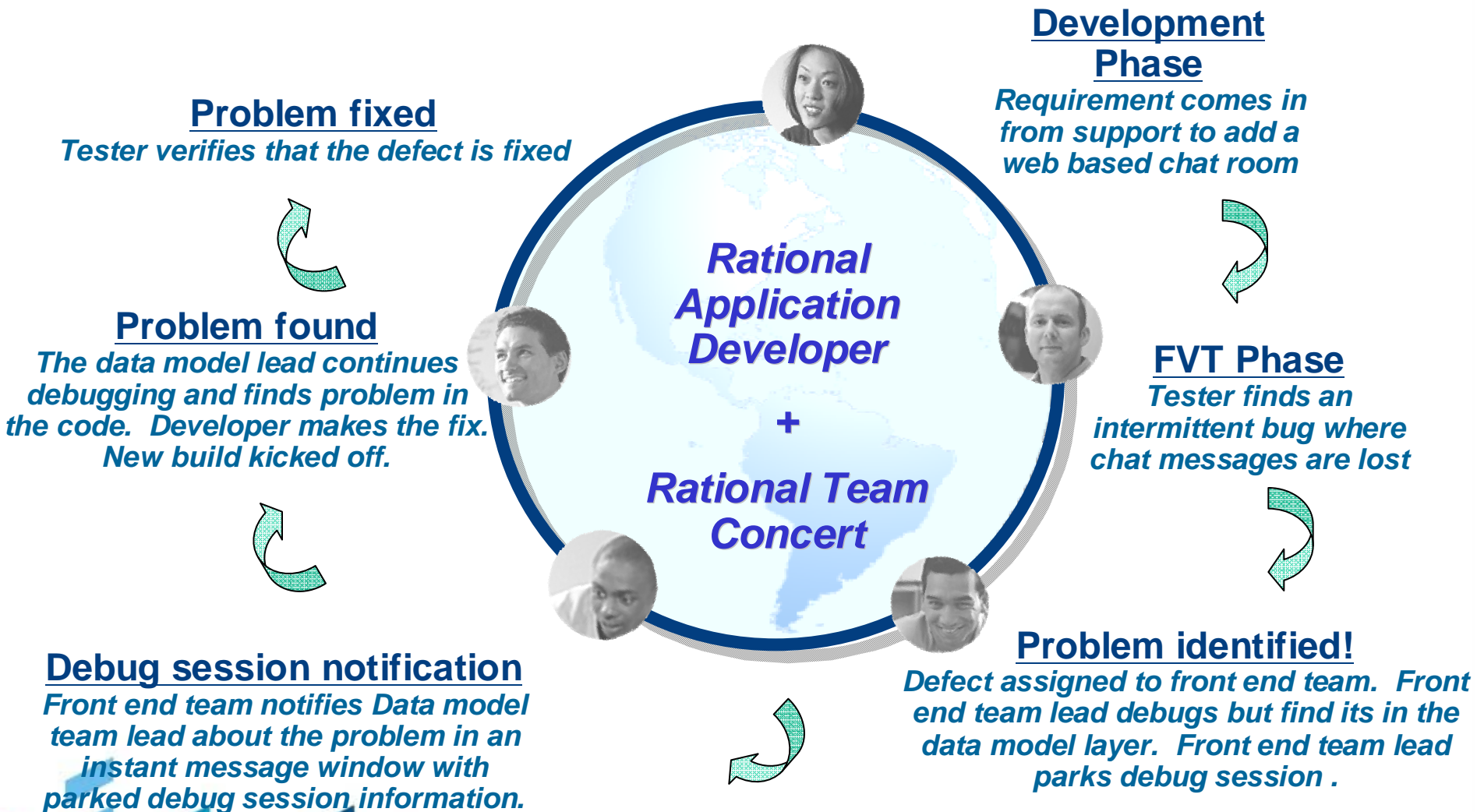
Customers.java 50%

Customers 50%

- incrementCapacity() : void 0%
- getCustomers() : Customer[] 100%
- Customers() 100%
- addCustomer(Customer) : void 75%

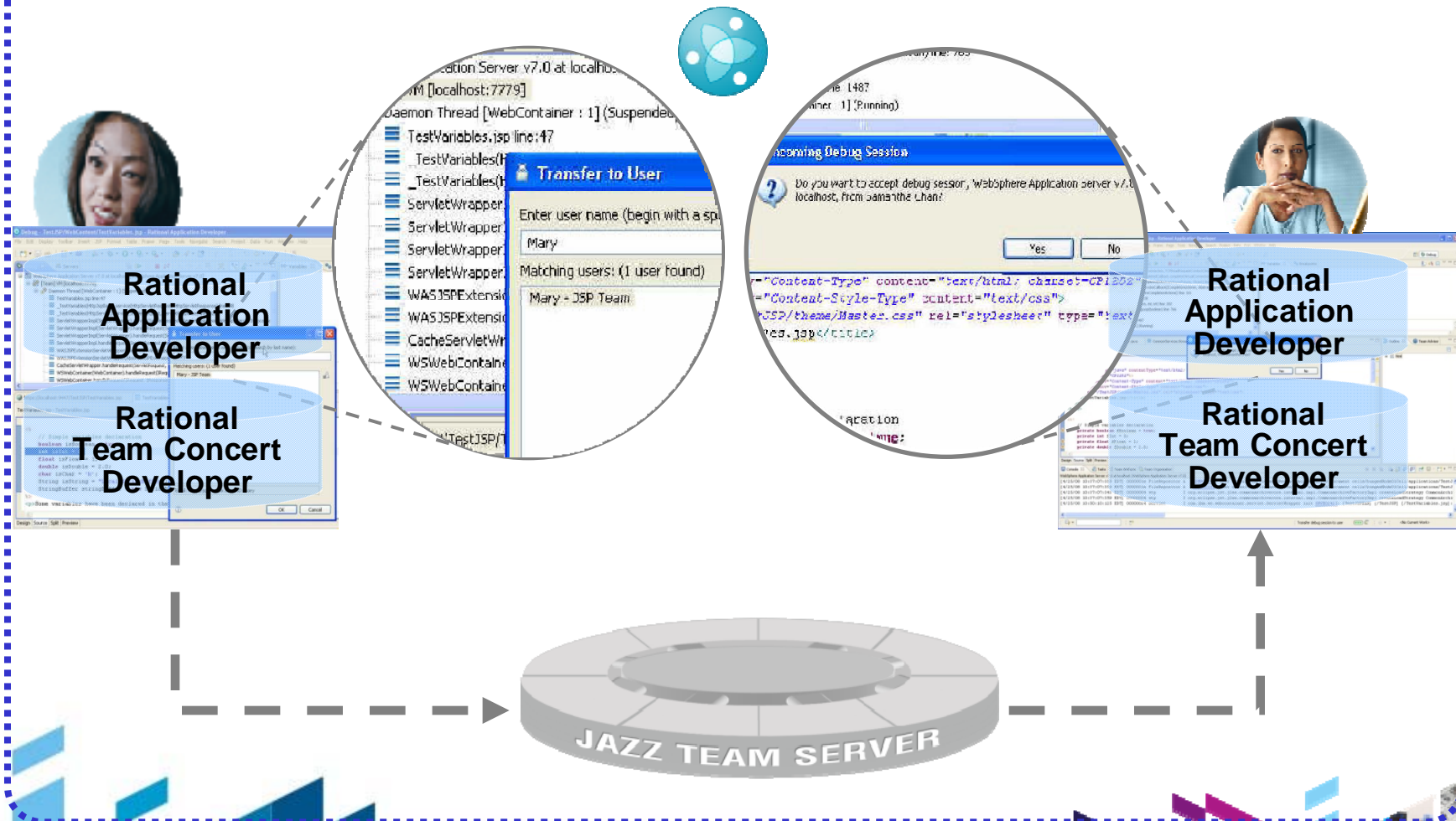
Paul.java	0%	0	3
Part.java	0%	0	8
Customers.java	50%	7	14
Customers	50%	7	14
incrementCapacity() : void	0%	0	6
getCustomers() : Customer[]	100%	1	1
Customers()	100%	3	3
addCustomer(Customer) : void	75%	3	4
Customer.java	100%	6	6

## Scenario #2 – Collaborative problem determination



Extend team collaboration to IBM middleware developers  
- resolve problems faster

## Sharing live debug sessions between team members





Now

Later

[riprasad@in.ibm.com](mailto:riprasad@in.ibm.com)

More Information

Rational® Application Developer Wiki



<http://www.ibm.com/developerworks/wikis/display/rad/Home>

# Innovate2010

The Rational Software Conference



# THANK YOU

[www.ibm.com/software/rational](http://www.ibm.com/software/rational)





