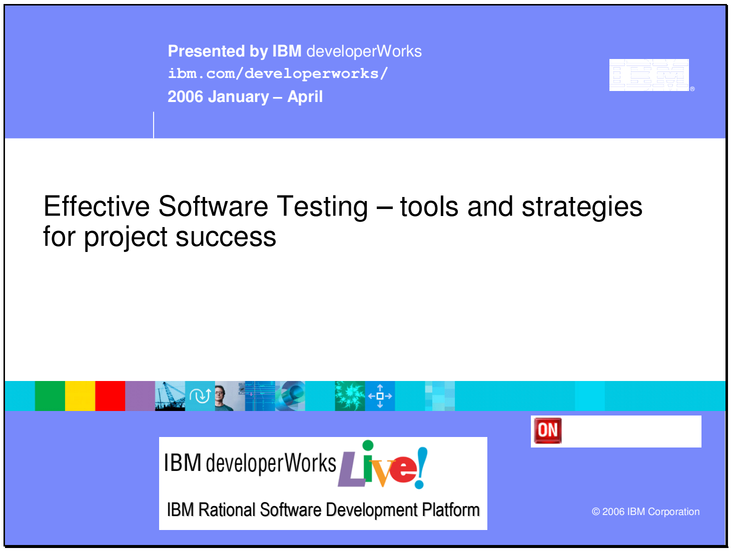
Slide 1



Presented by IBM developerWorks
`ibm.com/developerworks/`
2006 January – April

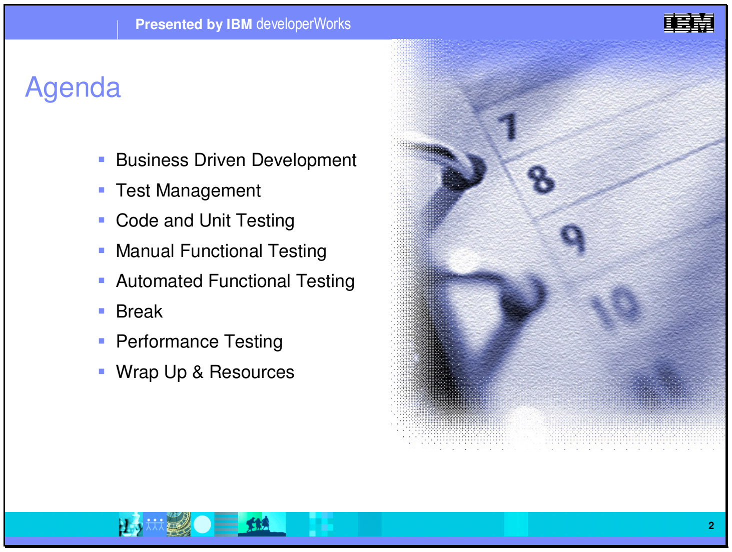
Effective Software Testing – tools and strategies for project success

ON

IBM developerWorks Live!
IBM Rational Software Development Platform

This paper is based on the developerWorks Live technical seminar Effective Software Testing – tools and strategies for project success.

Slide 2



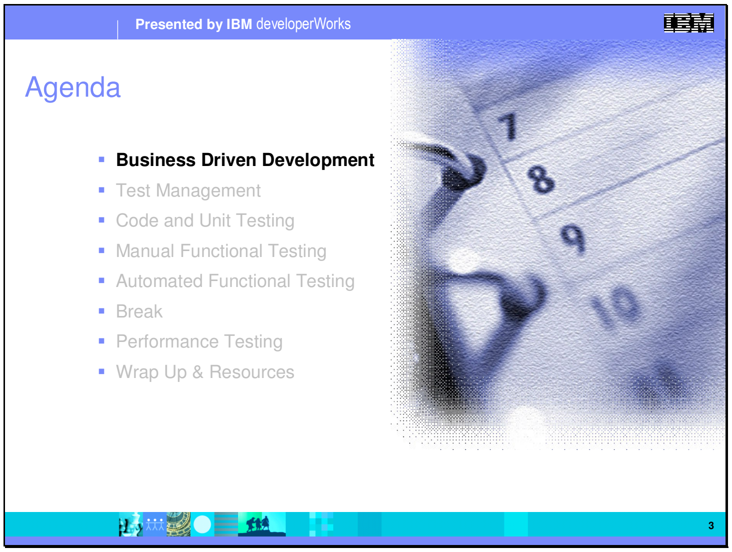
Presented by IBM developerWorks

Agenda

- Business Driven Development
- Test Management
- Code and Unit Testing
- Manual Functional Testing
- Automated Functional Testing
- Break
- Performance Testing
- Wrap Up & Resources

In this presentation, we will cover the lifecycle of software testing, beginning with the gathering and capturing of requirements in a process we call Business Driven Development, all the way to runtime performance testing. Our objective is to show why testing is becoming even more critical to the success of software projects, and why an integrated set of testing tools tied into tools for team, project, and requirements management can

Slide 3



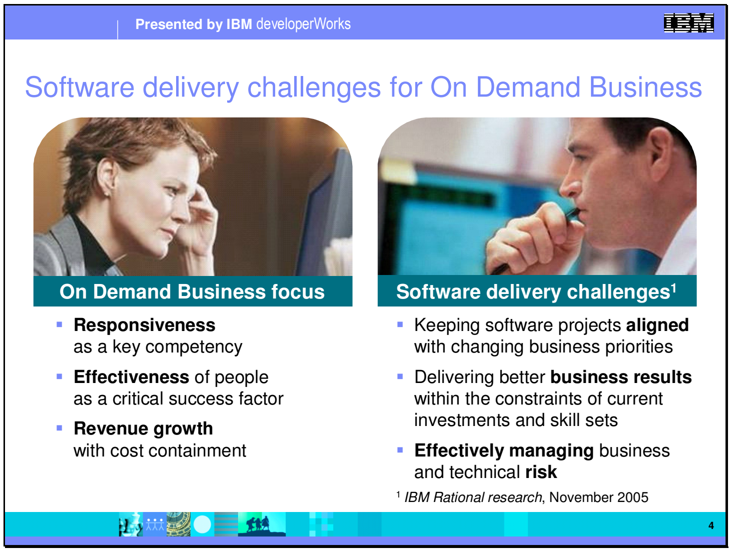
Presented by IBM developerWorks

Agenda

- **Business Driven Development**
- Test Management
- Code and Unit Testing
- Manual Functional Testing
- Automated Functional Testing
- Break
- Performance Testing
- Wrap Up & Resources

.

Slide 4



Presented by IBM developerWorks

Software delivery challenges for On Demand Business

On Demand Business focus

- **Responsiveness** as a key competency
- **Effectiveness** of people as a critical success factor
- **Revenue growth** with cost containment

Software delivery challenges[1]

- Keeping software projects **aligned** with changing business priorities
- Delivering better **business results** within the constraints of current investments and skill sets
- **Effectively managing** business and technical **risk**

[1] *IBM Rational research*, November 2005

4

Software is at the core of the transformation in business processes that must be made to compete successfully in the On Demand world.

Software is now:

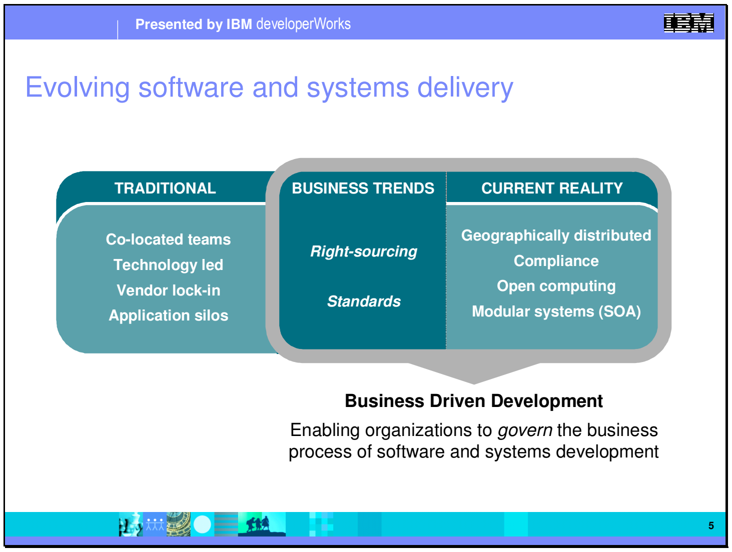
In the critical path of almost every product or project

The "smarts" and differentiators of most products and systems

It's no surprise, therefore, that CIOs, IT directors and managers tell us they are challenged with keeping IT aligned with the On Demand business priorities. They're challenged with improving the effectiveness of their project teams and existing technologies to deliver better business results. And, they are challenged with managing business and technical risk to lower the costs of the software innovation and change necessary for business resilience and growth.

The bottom line is that Global business trends are impacting both the type of business initiatives software delivery supports and well as the process of software delivery itself.

Slide 5



Presented by IBM developerWorks

Evolving software and systems delivery

TRADITIONAL

Co-located teams
Technology led
Vendor lock-in
Application silos

BUSINESS TRENDS

Right-sourcing

Standards

CURRENT REALITY

Geographically distributed
Compliance
Open computing
Modular systems (SOA)

Business Driven Development

Enabling organizations to *govern* the business process of software and systems development

5

Business of all sizes are enjoying whole new degrees of freedom in the way they perform work. Market forces and technology are leveling the playing field. Basically, any business activity or process that can be digitized can be sourced to the most knowledgeable workers, no matter the partner company they work for or where they are located in the world.   This focus on exploiting new opportunities in a world without boundaries is reflected in the software delivery challenges facing IT organizations today.

Business Trends:

First, we see a progression in how and where work is done – a work re-alignment. This is a change in focus from where work happens to how work can be aligned to:
- meet domestic markets and extend global reach
- and support customers and their experience as cost effectively as possible

Second, the evolution of right sourcing. Right sourcing models today have move from the outsourcing of non-critical IT functions to more critical business processes being distributed and performed by wherever most cost effective – sometimes by partners, sometimes by distributed work teams, sometimes by global system integrators – and this requires both great greater flexibility and greater control in the underlying software systems and the workforces that support such systems

Current Reality:

Geographically distributed teams needs to work together effectively to deliver high quality solution solutions.

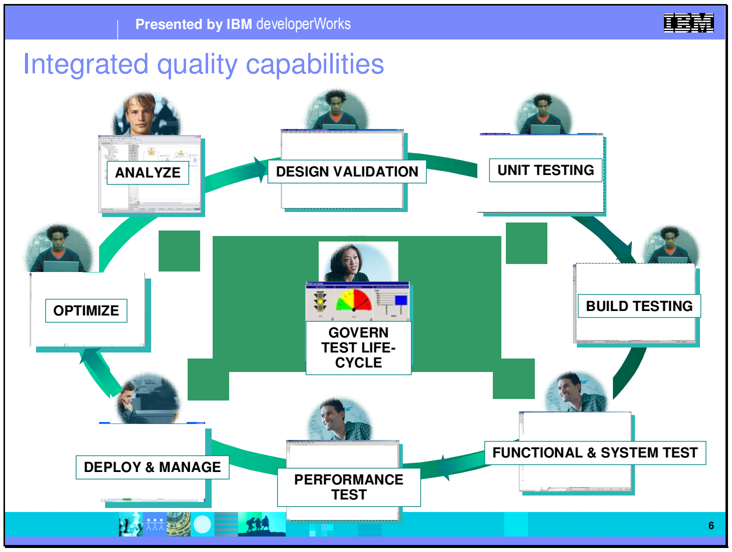
Distributed, multi-sourced business processes increases the complexity of compliance for both internal control and external standards and regulations. Compliance demands that you prove the processes are working and that you can validate the platform and deliverables.

Open computing breaks down walls among technologies, processes, and people to support information flow, process automation, collaboration, and innovation.

Development of modular systems enable businesses to leverage existing technologies in new ways and accelerate initiatives for faster time to value.

All of these trends impact not only development teams, but teams who are chartered with ensuring a high quality software system. These trends are what we call Business Driven Development – we are enabling business organizations to govern the business process of development across the entire lifecycle.

Slide 6



Presented by IBM developerWorks

Integrated quality capabilities

ANALYZE

DESIGN VALIDATION

UNIT TESTING

BUILD TESTING

FUNCTIONAL & SYSTEM TEST

PERFORMANCE TEST

DEPLOY & MANAGE

OPTIMIZE

GOVERN TEST LIFE-CYCLE

6

The full lifecycle of software development is continual flow of activities in a team process. Quality needs to be a factor from the very beginning, and threaded throughout the process. At each step, the team members will be performing the functions of the step, but need to be aware of the software quality aspects for that step – both what has happened in the previous step, and what will happen later. With an integrated set of tools supporting all aspects of the development process, team members have direct access to the quality elements created in previous steps, the quality elements being created by team members in the current step. For example:

During the analysis and requirements phase, you need to be considering testability

In the design phase, we are validating the proposed architectures for reliability along with the other design requirements

In development, Unit Testing plays a critical role in testing individual code units and components, and is the primary tool for doing build testing

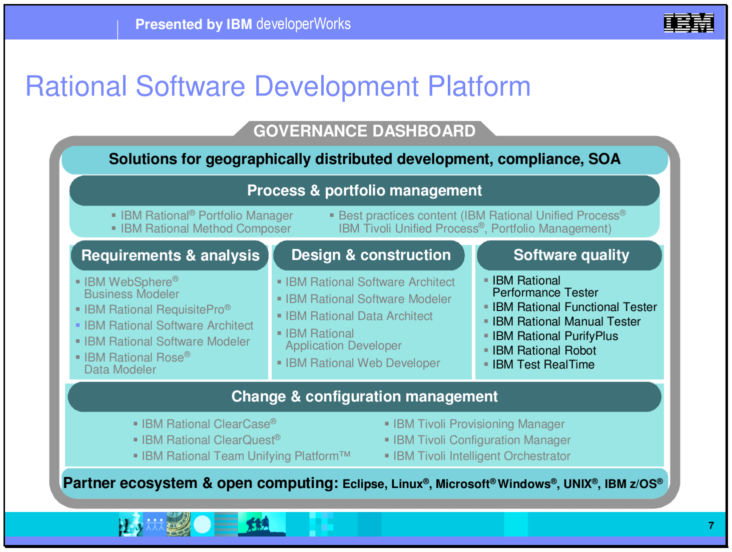
With a completed Build, we do functional testing – both regression testing for unmodified functions, and new functional testing for the added features. Automated testing, and tests tied to a particular version allow functionality to be verified quickly. New functional tests can be created for the new features, and the new version.

With completed functional testing, we can test applications to see if they are both scaleable, that they perform to the required level in the desired system platform.

Since many applications are deployed on multiple different platforms and sites, testing the application performance on a particular deployment platform is important. Things that are out of requirements compliance can be noted as change requests directly back into the configuration management tool. At this stage, it is important that system performance be continually monitored, and that the system be able to respond to changing usage patterns. Our Tivoli product line does this.

And finally, what has been learned and observed during the full process needs to be analyzed and feed back into the next development iteration.

Slide 7



Presented by IBM developerWorks

Rational Software Development Platform

GOVERNANCE DASHBOARD

Solutions for geographically distributed development, compliance, SOA

Process & portfolio management

- IBM Rational® Portfolio Manager
- IBM Rational Method Composer
- Best practices content (IBM Rational Unified Process® IBM Tivoli Unified Process®, Portfolio Management)

Requirements & analysis

- IBM WebSphere® Business Modeler
- IBM Rational RequisitePro®
- IBM Rational Software Architect
- IBM Rational Software Modeler
- IBM Rational Rose® Data Modeler

Design & construction

- IBM Rational Software Architect
- IBM Rational Software Modeler
- IBM Rational Data Architect
- IBM Rational Application Developer
- IBM Rational Web Developer

Software quality

- IBM Rational Performance Tester
- IBM Rational Functional Tester
- IBM Rational Manual Tester
- IBM Rational PurifyPlus
- IBM Rational Robot
- IBM Test RealTime

Change & configuration management

- IBM Rational ClearCase®
- IBM Rational ClearQuest®
- IBM Rational Team Unifying Platform™
- IBM Tivoli Provisioning Manager
- IBM Tivoli Configuration Manager
- IBM Tivoli Intelligent Orchestrator

Partner ecosystem & open computing: Eclipse, Linux®, Microsoft® Windows®, UNIX®, IBM z/OS®

7

The Rational Software Development Platform has tools covering all aspects of software quality, and is surrounded by what we call a “Governance Dashboard”.

In Process and portfolio management, we are tied into the business itself, understanding what the business requirements and priorities are.

In requirements and analysis, we have tools for creating and maintaining the requirements along with the ability to tie these requirements to code, tests, and activities.

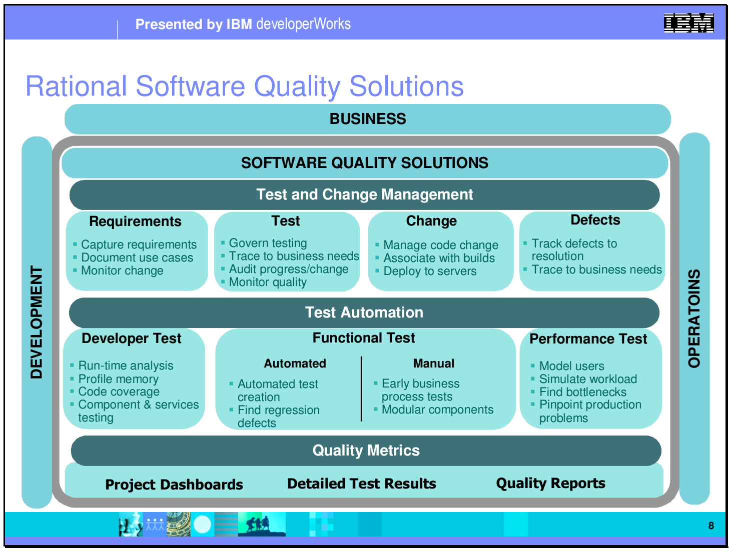
In design and construction, we have a set of tools that allow us to build the application; from architecting, modeling, to code creation.

In software quality, we have the tools which focus on this aspect ranging from component and code analysis with Purify, full system and real-time testing.

Underlying all of this is a set of tools for change and configuration management – insuring we are following the correct processes, insuring those processes are auditable and compliant, and version control of the software assets.

Rational ClearQuest will tie all of these tools together and give a management and governance platform.

Slide 8



Presented by IBM developerWorks

Rational Software Quality Solutions

BUSINESS

DEVELOPMENT

OPERATOINS

SOFTWARE QUALITY SOLUTIONS

Test and Change Management

Requirements
- Capture requirements
- Document use cases
- Monitor change

Test
- Govern testing
- Trace to business needs
- Audit progress/change
- Monitor quality

Change
- Manage code change
- Associate with builds
- Deploy to servers

Defects
- Track defects to resolution
- Trace to business needs

Test Automation

Developer Test
- Run-time analysis
- Profile memory
- Code coverage
- Component & services testing

Functional Test

Automated
- Automated test creation
- Find regression defects

Manual
- Early business process tests
- Modular components

Performance Test
- Model users
- Simulate workload
- Find bottlenecks
- Pinpoint production problems

Quality Metrics

Project Dashboards | Detailed Test Results | Quality Reports

8

From a software quality perspective, there are three aspects which are important.

Test and Change Management

We manage requirements, and create the tests for those requirements. From this, we manage the change requests as well as the identified defects, rolling them back into the requirements process for the next version.

Test Automation

Automation is an important part of overall quality. Automated testing tools along with the automation that comes from an understood and agreed upon process are the two main components – they provide reproducibility both with manual steps such as unit testing or manual testing, and automated tests tied to specific versions of the application.

Quality Metrics

Management of the overall process is critical. Different views of the state of a project – dashboards, test results, and reports - are needed both for business management and project management.

Slide 9

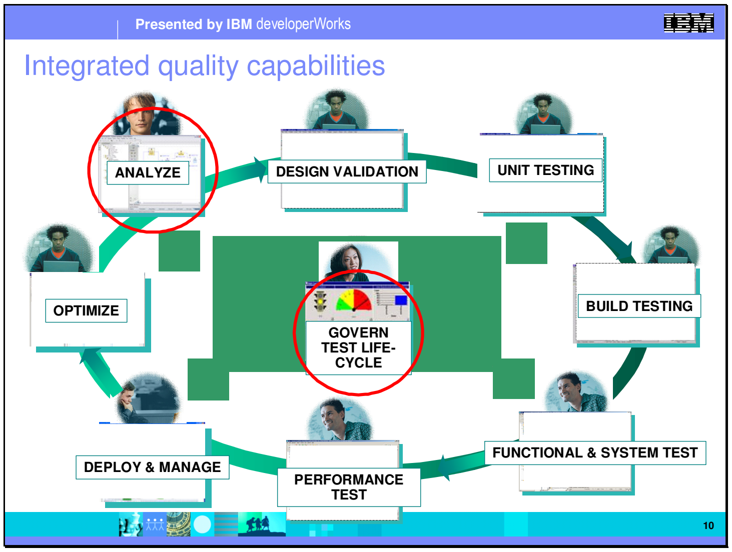
Presented by **IBM** developerWorks

Agenda

- Business Driven Development
- **Test Management**
- Code and Unit Testing
- Manual Functional Testing
- Automated Functional Testing
- Break
- Performance Testing
- Wrap Up & Resources

.

Slide 10

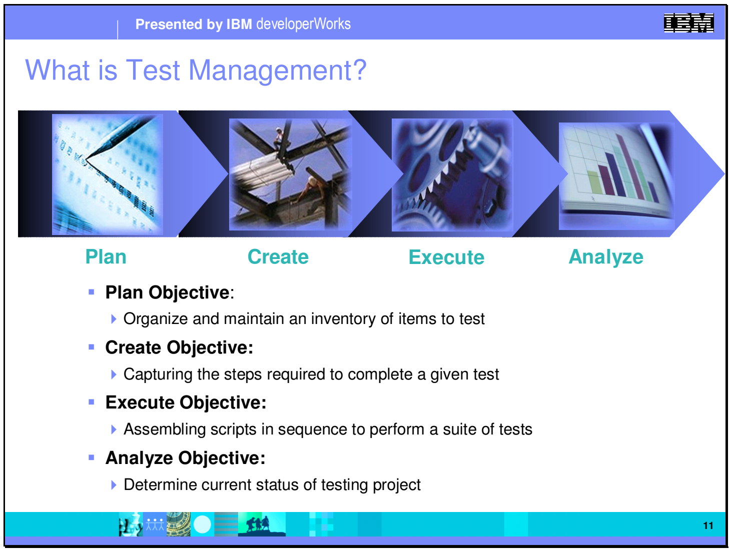


Presented by IBM developerWorks

Integrated quality capabilities

The first aspect of software quality we will focus on is the governance of the application lifecycle, and the management of the requirements which define what we are doing. The tool we will be primarily using is ClearQuest and the Test Manager

Slide 11



Presented by **IBM** developerWorks

What is Test Management?

Plan
Create
Execute
Analyze

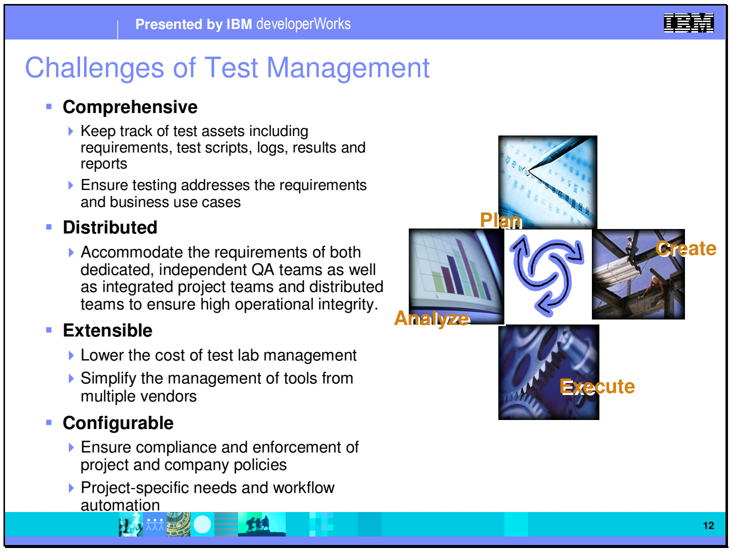
- **Plan Objective**:
  - Organize and maintain an inventory of items to test
- **Create Objective:**
  - Capturing the steps required to complete a given test
- **Execute Objective:**
  - Assembling scripts in sequence to perform a suite of tests
- **Analyze Objective:**
  - Determine current status of testing project

So, what is Test Management? The objective here is simple: determine the current status of the testing project.

Key to consider here are the ability to get a quick report on test status. You'll want to be able to know where you are at any given point – especially when the project manager comes calling in a panic. You'll also want to be able to keep up with underlying requirement changes.

Requirement, or motivator changes pose a great problem for testers. If a requirement changes, is the associated test still valid? That's a decision that you'll have to make – but you'll only be able to make that decision if you know the requirement changes! TestManager will let you know.

Slide 12


Plan
Create
Analyze
Execute

Presented by IBM developerWorks

Challenges of Test Management

- **Comprehensive**
  - Keep track of test assets including requirements, test scripts, logs, results and reports
  - Ensure testing addresses the requirements and business use cases
- **Distributed**
  - Accommodate the requirements of both dedicated, independent QA teams as well as integrated project teams and distributed teams to ensure high operational integrity.
- **Extensible**
  - Lower the cost of test lab management
  - Simplify the management of tools from multiple vendors
- **Configurable**
  - Ensure compliance and enforcement of project and company policies
  - Project-specific needs and workflow automation

If we look at the challenges of test management, there are four principal areas of challenge.

Comprehensiveness

Test management efforts need to be comprehensive. We need to keep track of all test assets created – scripts, test logs, results, reports – everything related to the test effort. We also need to know that we are covering all of the requirements, and be able to demonstrate this coverage by linking back tests to particular requirements.

Distributed Teams

Test management needs to respond to the global nature of many development projects. Everyone on the development, testing, and project management teams needs access to these test assets, so they need to be distributed and accessible geographically.

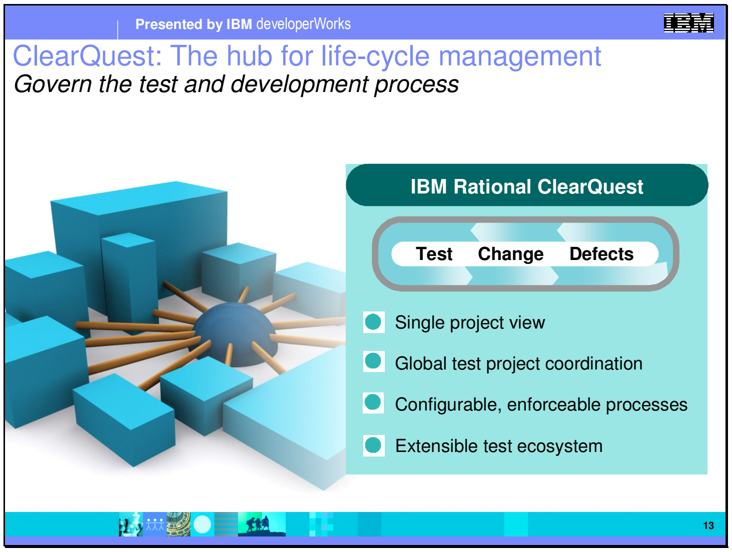
Extensibility of the testing tools

To lower the cost of the overall testing effort, we need a test platform that is integrated and extensible for all the testing efforts. Stand-alone testing tools can break this integrated linkage. With the Eclipse platform, specialized testing tools from different venders can be integrated into the full development environment.

Configurability of the testing tools

Finally, our testing tools need to be configurable to respond to changes in process, and changes in external regulations such as individual sign-off of tests as part of an audit trail using electronic signatures.

Slide 13



Presented by IBM developerWorks

ClearQuest: The hub for life-cycle management

Govern the test and development process

IBM Rational ClearQuest

Test Change Defects

- Single project view
- Global test project coordination
- Configurable, enforceable processes
- Extensible test ecosystem

13

The tool that we provide for test management is Rational ClearQuest. You can think of this as the central hub both for test management and overall project management. If we look at how ClearQuest responds to the four challenges, we have;

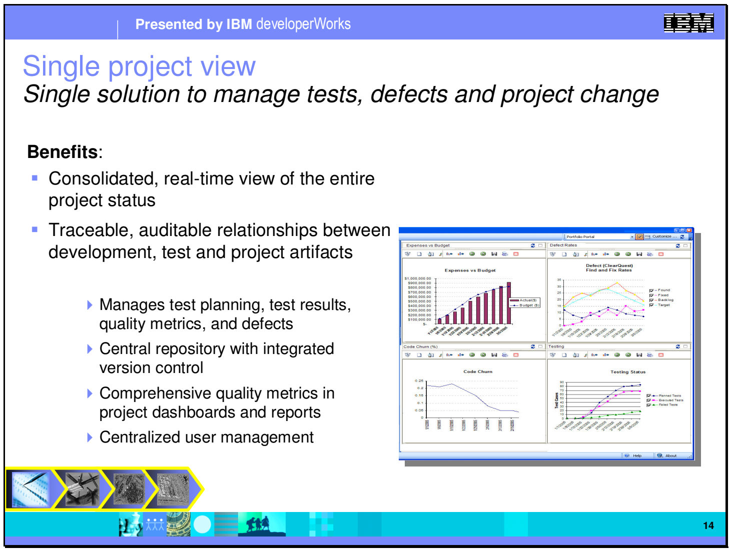
A single project view. You can see all of your test assets and projects from a single view.

Global coordination of both the testing and the project. Test and project assets can be maintained in a single repository, or across multiple replicated repositories.

Configurable and enforceable processes. Development processes can be fully customized, and it comes with and electronic signature package to support test auditing.

Extensible test ecosystem. The platform that the Rational products are based on allows for the integration of other tests into the development tool. This comes from the Eclipse framework which we will talk about later.

Slide 14



Presented by **IBM** developerWorks

Single project view

Single solution to manage tests, defects and project change

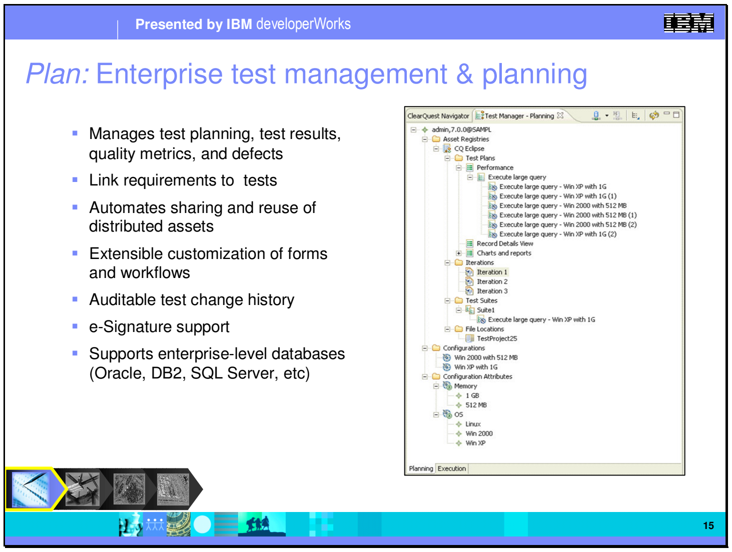
Benefits:

- Consolidated, real-time view of the entire project status
- Traceable, auditable relationships between development, test and project artifacts
  - Manages test planning, test results, quality metrics, and defects
  - Central repository with integrated version control
  - Comprehensive quality metrics in project dashboards and reports
  - Centralized user management

The single project view that Rational ClearQuest gives not only gives a single view of testing, but also of the full project. Requirements, change requests, and defects can be tracked through the development process and into the testing process. Configurable views and reports of the project and testing allow managers to understand the state and health of the development project. This can significantly reduce the risks associated with software development projects.

ClearQuest, along with ClearCase, provide full versioning of this process along with the assets associated with a particular version. Together they form the center of configuration management and project control.

Slide 15



Presented by IBM developerWorks

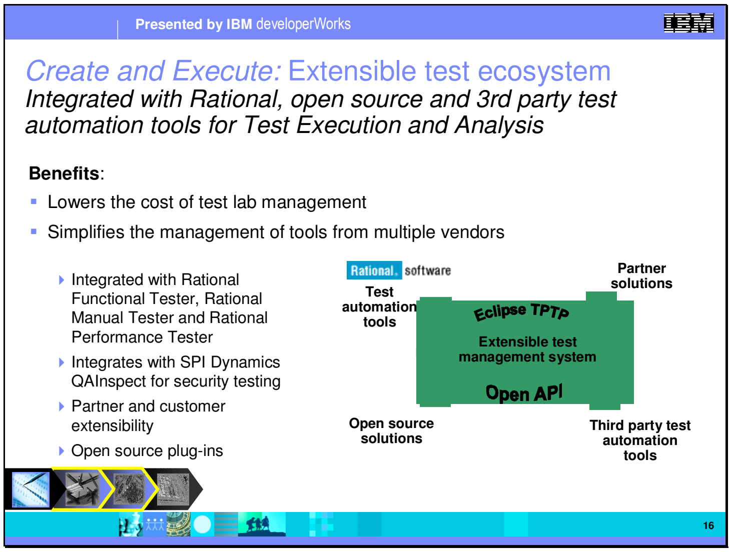
Plan: Enterprise test management & planning

- Manages test planning, test results, quality metrics, and defects
- Link requirements to tests
- Automates sharing and reuse of distributed assets
- Extensible customization of forms and workflows
- Auditable test change history
- e-Signature support
- Supports enterprise-level databases (Oracle, DB2, SQL Server, etc)

```
admin,7.0.0@SAMPL
  Asset Registries
    CQ Eclipse
      Test Plans
        Performance
          Execute large query
            Execute large query - Win XP with 1G
            Execute large query - Win XP with 1G (1)
            Execute large query - Win 2000 with 512 MB
            Execute large query - Win 2000 with 512 MB (1)
            Execute large query - Win 2000 with 512 MB (2)
            Execute large query - Win XP with 1G (2)
        Record Details View
        Charts and reports
      Iterations
        Iteration 1
        Iteration 2
        Iteration 3
      Test Suites
        Suite1
          Execute large query - Win XP with 1G
      File Locations
        TestProject25
  Configurations
    Win 2000 with 512 MB
    Win XP with 1G
  Configuration Attributes
    Memory
      1 GB
      512 MB
    OS
      Linux
      Win 2000
      Win XP
```

Out of the box, Rational ClearQuest has all the features you need for effective test management and planning. But most important, is that the internal view of software development process – Rational Unified Process (RUP) – is fully configurable to conform to your own internal process. This customized process can then be used with all the integrated development tools to help automate the production of quality code and applications.

Slide 16



Presented by IBM developerWorks

Create and Execute: Extensible test ecosystem

Integrated with Rational, open source and 3rd party test automation tools for Test Execution and Analysis

Benefits:

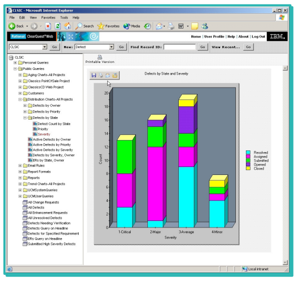
- Lowers the cost of test lab management
- Simplifies the management of tools from multiple vendors
  - Integrated with Rational Functional Tester, Rational Manual Tester and Rational Performance Tester
  - Integrates with SPI Dynamics QAInspect for security testing
  - Partner and customer extensibility
  - Open source plug-ins

In creating and executing tests, the Rational test tools can be extended either with internally developed test tools, or third party tool such as SPI Dynamics and QAInspect. This is possible because all of the Rational development tools are built on top of the Eclipse Framework. Eclipse is an open source framework for tools development. In testing, we take advantage of the Eclipse TPTP – Test and Performance Tools Platform. This provides us links to a common API allowing integration with third party tools that are also built on Eclipse. This extensibility gives you access to the testing tools you need, and at the same time lowering overall cost for testing.

Slide 17

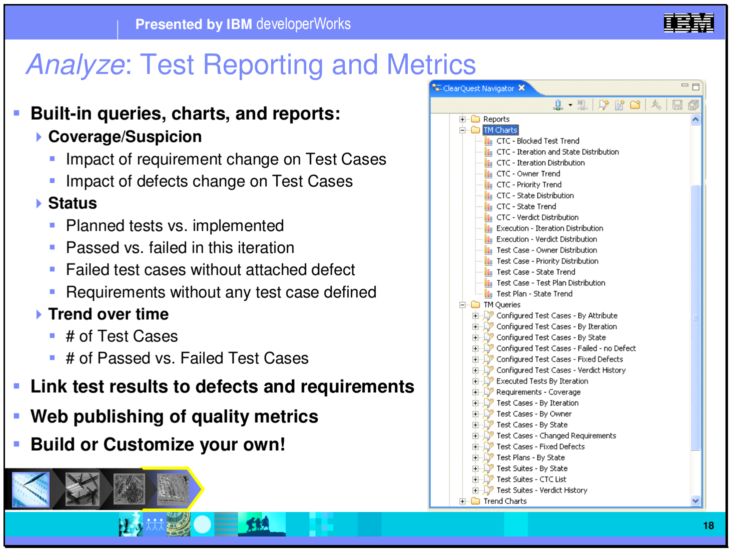
Analyze: Comprehensive quality metrics

- Generate reports to monitor test progress
  - Run reports to summarize test case results
  - Run reports to measure requirements coverage
- Key considerations when creating schedules
  - Ability to report against test motivators
  - Ability to monitor current project status



Inside ClearQuest, there are a number of prepared report formats out of the box, you are free to customize the charts, metrics and reports you need to manage project success. These reports can be automated, and ad hoc.

Slide 18



Presented by IBM developerWorks

Analyze: Test Reporting and Metrics

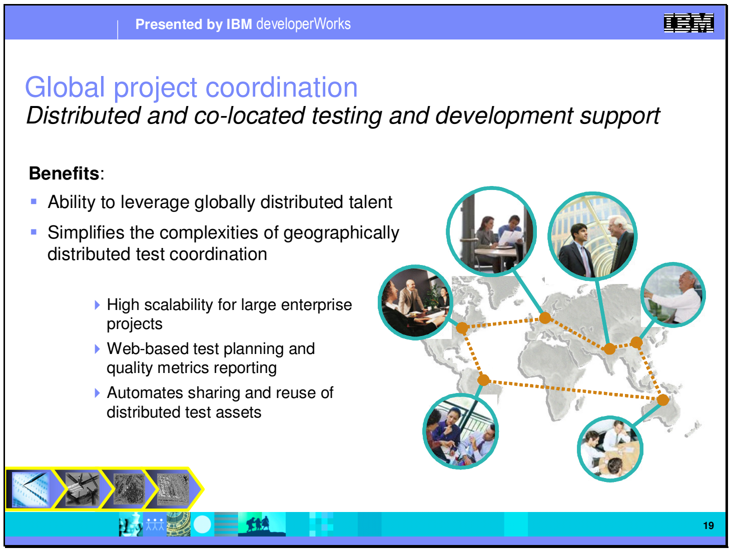
- **Built-in queries, charts, and reports:**
  - **Coverage/Suspicion**
    - Impact of requirement change on Test Cases
    - Impact of defects change on Test Cases
  - **Status**
    - Planned tests vs. implemented
    - Passed vs. failed in this iteration
    - Failed test cases without attached defect
    - Requirements without any test case defined
  - **Trend over time**
    - # of Test Cases
    - # of Passed vs. Failed Test Cases
- **Link test results to defects and requirements**
- **Web publishing of quality metrics**
- **Build or Customize your own!**

ClearQuest Navigator

- Reports
- TM Charts
  - CTC - Blocked Test Trend
  - CTC - Iteration and State Distribution
  - CTC - Iteration Distribution
  - CTC - Owner Trend
  - CTC - Priority Trend
  - CTC - State Distribution
  - CTC - State Trend
  - CTC - Verdict Distribution
  - Execution - Iteration Distribution
  - Execution - Verdict Distribution
  - Test Case - Owner Distribution
  - Test Case - Priority Distribution
  - Test Case - State Trend
  - Test Case - Test Plan Distribution
  - Test Plan - State Trend
- TM Queries
  - Configured Test Cases - By Attribute
  - Configured Test Cases - By Iteration
  - Configured Test Cases - By State
  - Configured Test Cases - Failed - no Defect
  - Configured Test Cases - Fixed Defects
  - Configured Test Cases - Verdict History
  - Executed Tests By Iteration
  - Requirements - Coverage
  - Test Cases - By Iteration
  - Test Cases - By Owner
  - Test Cases - By State
  - Test Cases - Changed Requirements
  - Test Cases - Fixed Defects
  - Test Plans - By State
  - Test Suites - By State
  - Test Suites - CTC List
  - Test Suites - Verdict History
- Trend Charts

Above are some examples of the kinds of test reporting and the underlying metrics that are available out of the box. These tests are based on the Rational Unified Process model and are fully configurable. Additionally, all of the reports are available through a web based client. This gives access to test results and metrics to all interested parties.

Slide 19



Presented by **IBM** developerWorks

Global project coordination

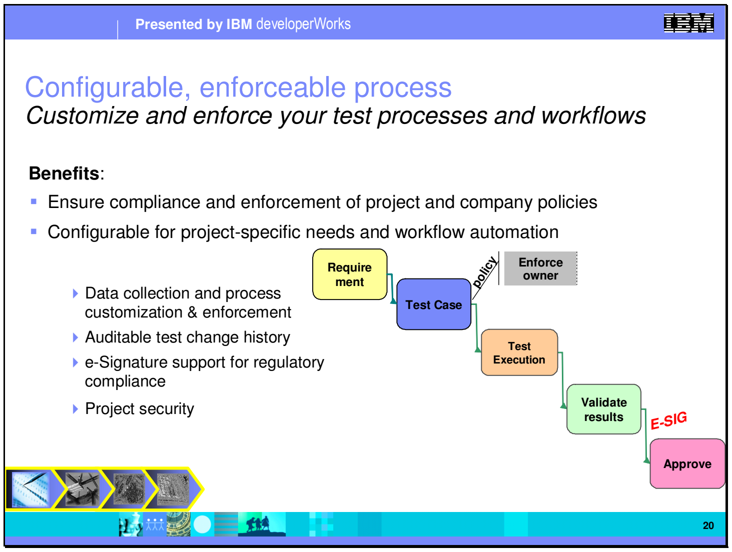
Distributed and co-located testing and development support

Benefits:

- Ability to leverage globally distributed talent
- Simplifies the complexities of geographically distributed test coordination
  - High scalability for large enterprise projects
  - Web-based test planning and quality metrics reporting
  - Automates sharing and reuse of distributed test assets

ClearQuest, under the covers, simplifies the complexity of geographically distributed development teams. This gives access to testing personnel and test assets regardless of their location. This can be done via a single central data repository, or a highly scaleable ClearQuest multi-site version with full replication.

Slide 20



Presented by IBM developerWorks

Configurable, enforceable process

Customize and enforce your test processes and workflows

Benefits:

- Ensure compliance and enforcement of project and company policies
- Configurable for project-specific needs and workflow automation
  - Data collection and process customization & enforcement
  - Auditable test change history
  - e-Signature support for regulatory compliance
  - Project security

Require ment

Test Case

policy

Enforce owner

Test Execution

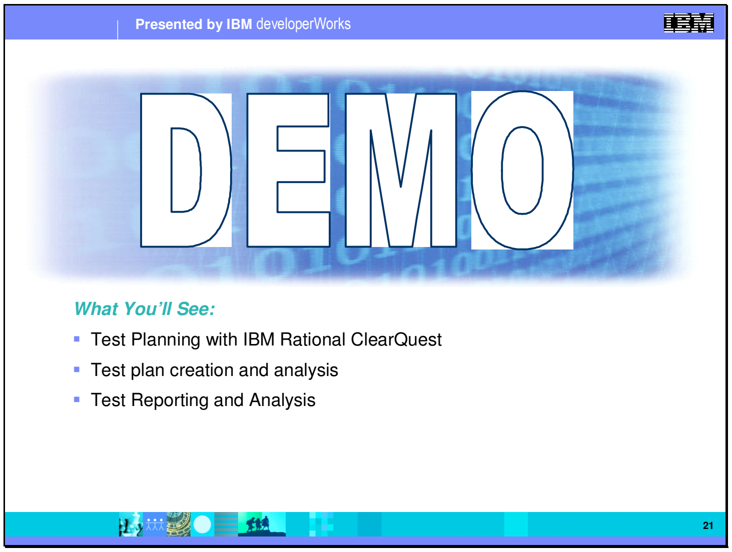
Validate results

E-SIG

Approve

We mentioned before that automation of processes helps reduce cost, and increase quality. To get the full benefits of this in a development project, the processes need to be part of the tools themselves. This helps to “automate” compliance to the process, and can give a clear audit trail for the full development lifecycle. For this to be effective though, the process enforced by the tools needs to be configurable to reflect the real development processes used. It also needs to be able to change to respond new business and external governance rules. A good example of this is the use of electronic signatures to show sign-off at various key stages in the process – testing, validation of results, approval, etc.

Slide 21

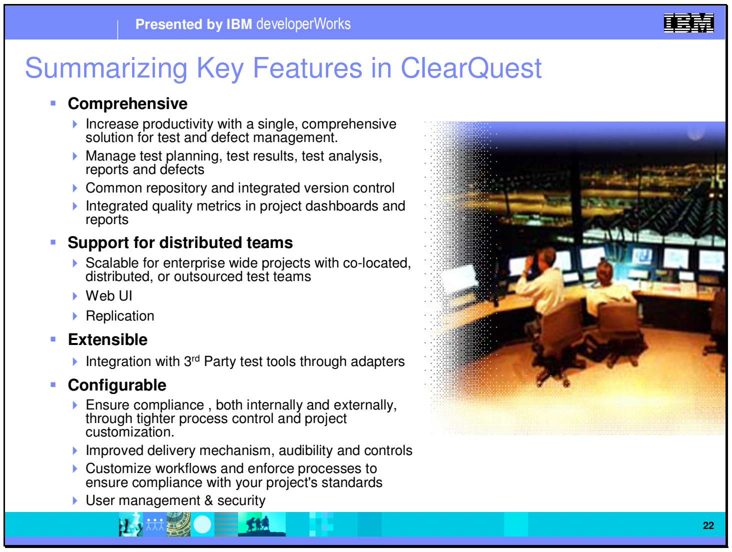


DEMO

What You'll See:

- Test Planning with IBM Rational ClearQuest
- Test plan creation and analysis
- Test Reporting and Analysis

Slide 22



Presented by IBM developerWorks

Summarizing Key Features in ClearQuest

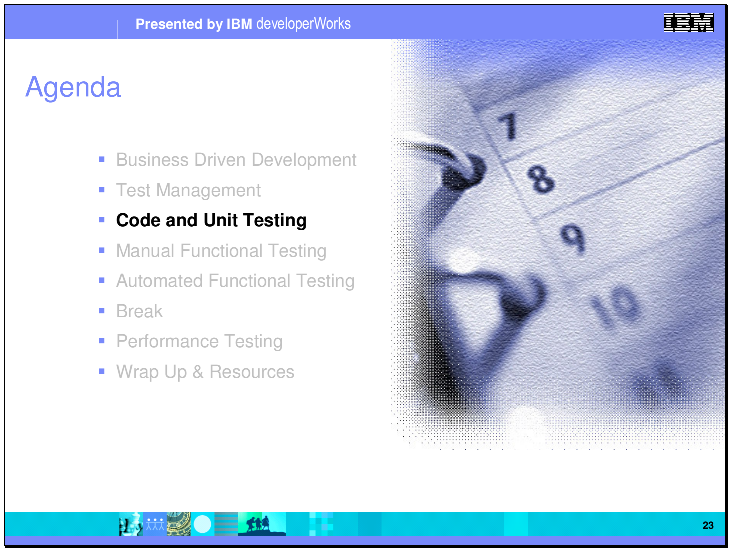
- **Comprehensive**
  - Increase productivity with a single, comprehensive solution for test and defect management.
  - Manage test planning, test results, test analysis, reports and defects
  - Common repository and integrated version control
  - Integrated quality metrics in project dashboards and reports
- **Support for distributed teams**
  - Scalable for enterprise wide projects with co-located, distributed, or outsourced test teams
  - Web UI
  - Replication
- **Extensible**
  - Integration with 3rd Party test tools through adapters
- **Configurable**
  - Ensure compliance , both internally and externally, through tighter process control and project customization.
  - Improved delivery mechanism, audibility and controls
  - Customize workflows and enforce processes to ensure compliance with your project's standards
  - User management & security

Hopefully we’ve given you a realistic view of what you can expect as you take your first step towards automated test asset management. Let’s review some of the keys to delivering quality on time…

Crucial is the ability to support all types of testing…performance, functional and manual. By automating all phases of the testing life cycle, from planning through execution to data analysis, the tools will ensure your test group runs at maximum efficiency.

Finally, the tools enable you to effortlessly track your testing progress. By taking care of all the data gathering and reporting for you, ClearQuest can instantly keep you up to date on test progress reporting, requirement changes and configuration testing … to name a few.

Slide 23



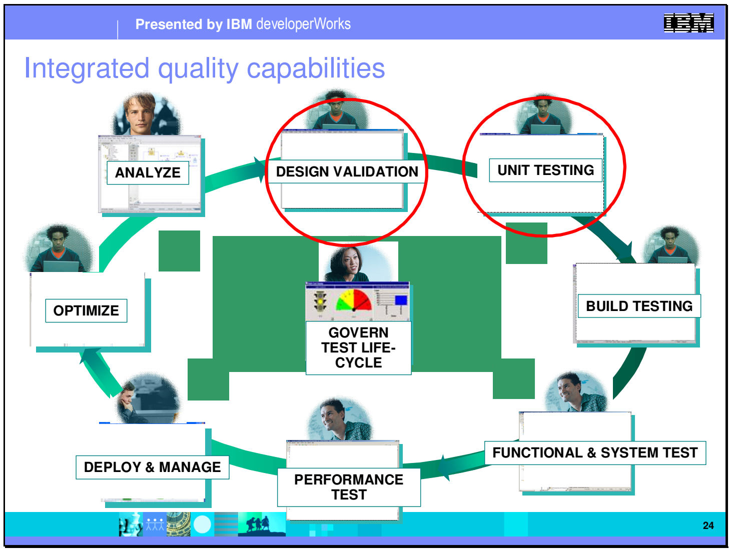
Presented by IBM developerWorks

Agenda

- Business Driven Development
- Test Management
- **Code and Unit Testing**
- Manual Functional Testing
- Automated Functional Testing
- Break
- Performance Testing
- Wrap Up & Resources

In this section, we will look at the testing that goes on during the development process – Code and Unit Testing.

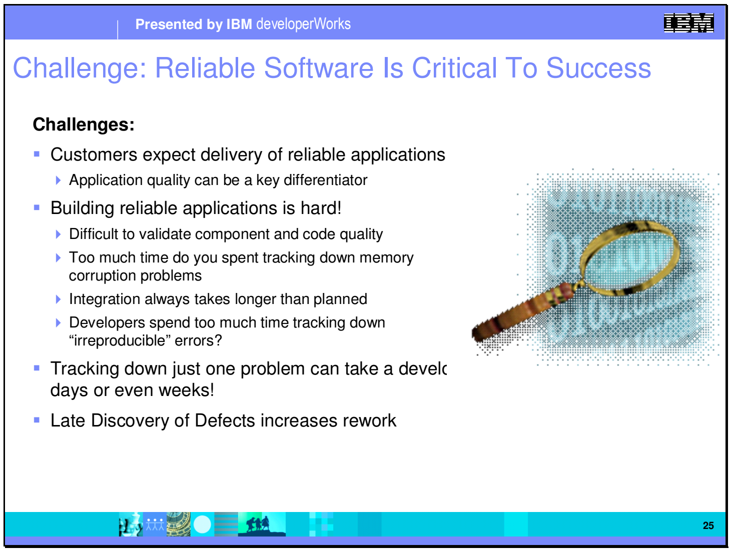
Slide 24



Integrated quality capabilities

In focusing on improving software quality, software engineering techniques have been pushing more and more of the testing into an earlier stage. In this section we are going to look at how this affects the architecture, design, and coding stages.

Slide 25



Presented by IBM developerWorks

Challenge: Reliable Software Is Critical To Success

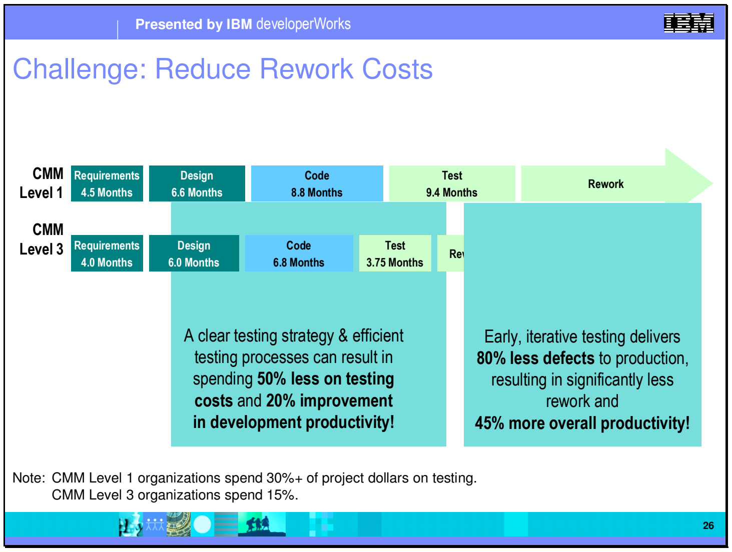
Challenges:

- Customers expect delivery of reliable applications
  - Application quality can be a key differentiator
- Building reliable applications is hard!
  - Difficult to validate component and code quality
  - Too much time do you spent tracking down memory corruption problems
  - Integration always takes longer than planned
  - Developers spend too much time tracking down “irreproducible” errors?
- Tracking down just one problem can take a develc days or even weeks!
- Late Discovery of Defects increases rework

25

Its clear that software quality is critical to success. The connection between how a company does business and the underlying IT system is direct. But adding quality into development at an early stage is a hard problem. With a focus on delivering code on schedule, chasing down problems can take a developer days and even weeks. It is easy to justify differing problem resolution to later stages, but this only makes the problem worse. Late discovery of defects increases rework, and costs.

Slide 26



Presented by IBM developerWorks

Challenge: Reduce Rework Costs

CMM Level 1

Requirements 4.5 Months

Design 6.6 Months

Code 8.8 Months

Test 9.4 Months

Rework

CMM Level 3

Requirements 4.0 Months

Design 6.0 Months

Code 6.8 Months

Test 3.75 Months

Rev

A clear testing strategy & efficient testing processes can result in spending **50% less on testing costs** and **20% improvement in development productivity!**

Early, iterative testing delivers **80% less defects** to production, resulting in significantly less rework and **45% more overall productivity!**

Note: CMM Level 1 organizations spend 30%+ of project dollars on testing.
CMM Level 3 organizations spend 15%.

26

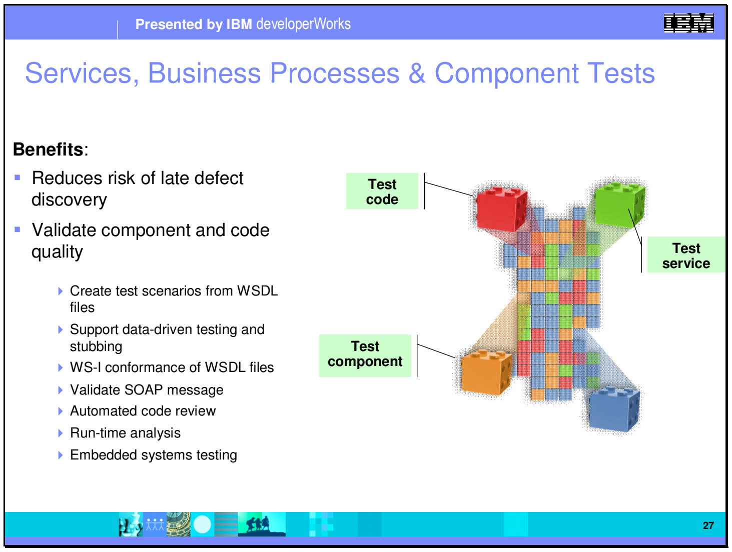
SEI (Software Engineering Institute) claims that CMMi level 3 projects will show around 20% productivity increase. This requires institutionalization of common processes, methods and tools across an organization with a common line of business, plus objective metrics, plus maturity in best practices gets an organization to improving return on investment from project to project.

This benefit is mostly visible during the testing activities: defining a clear testing strategy and implementing an efficient testing process can drive to a 50% decrease of the global testing costs.

More, with earlier and smarter tests, the applications are delivered with 80% bug less, reducing the rework (working on residual bugs) and the overall productivity is increased by 45%.

That’s why the testing strategy must be the concern of all the organization, at all levels.

Slide 27



Presented by IBM developerWorks

Services, Business Processes & Component Tests

Benefits:

- Reduces risk of late defect discovery
- Validate component and code quality
  - Create test scenarios from WSDL files
  - Support data-driven testing and stubbing
  - WS-I conformance of WSDL files
  - Validate SOAP message
  - Automated code review
  - Run-time analysis
  - Embedded systems testing

Test code

Test service

Test component

27

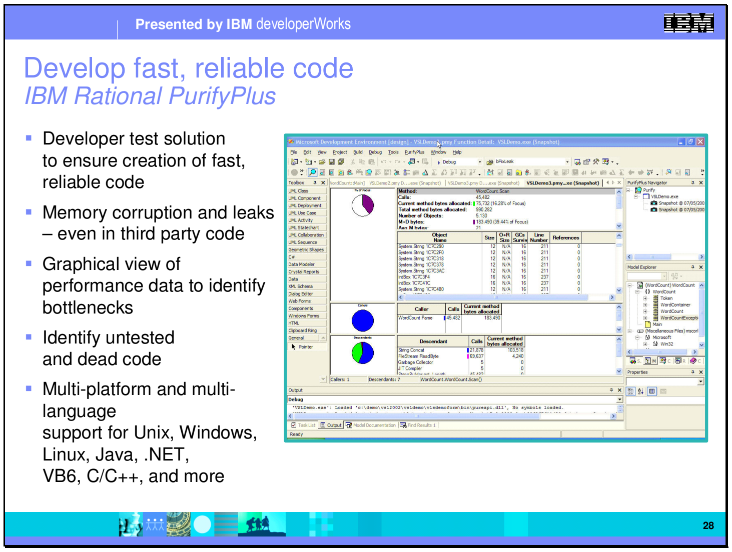
Opportunity:

Modular systems are the most cost-effective way to provide the flexible, agile systems demanded by On Demand Business. They enable businesses to “deconstruct” critical business processes into components that can then be “reconstructed” into new processes more quickly. Modular systems are also based on open computing to maximize interoperability. And they are scalable. As a result businesses can react rapidly to new customer demands and market opportunities, as well as competitive threats.

To ensure rapid testing of modular systems, Rational provides customers with a multi-layer approach to testing. The key benefit is to identify defects at an early stage which reduces the overall costs.

During development, we are interested in such things as; automated unit testing – JUnit, testing of web services using published WSDL definitions, data driven testing, automated code reviews, static and runtime analysis, etc.

Slide 28



Presented by IBM developerWorks

Develop fast, reliable code

IBM Rational PurifyPlus

- Developer test solution to ensure creation of fast, reliable code
- Memory corruption and leaks – even in third party code
- Graphical view of performance data to identify bottlenecks
- Identify untested and dead code
- Multi-platform and multi-language support for Unix, Windows, Linux, Java, .NET, VB6, C/C++, and more

So to develop fast, reliable code, you need a tool that automatically pinpoints hard-to-find bugs, highlights performance bottlenecks, and helps you avoid shipping untested code, I.e. you need Rational PurifyPlus.

It combines the strengths of three industry leading products in one box:

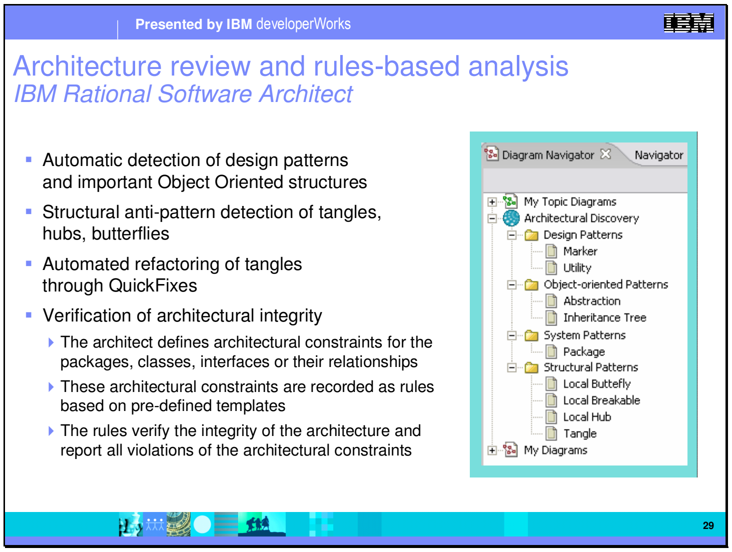
Rational Purify – memory leaks, memory corruption,

Rational Quantify – performance bottlenecks

Rational PureCoverage – see line by line the code coverage for tests

Purify Plus supports multiple languages, and multiple platforms Unix, Windows, Linux, Java, .NET, VB6, C/C++, and more.

Slide 29



Presented by IBM developerWorks

Architecture review and rules-based analysis

IBM Rational Software Architect

- Automatic detection of design patterns and important Object Oriented structures
- Structural anti-pattern detection of tangles, hubs, butterflies
- Automated refactoring of tangles through QuickFixes
- Verification of architectural integrity
  - The architect defines architectural constraints for the packages, classes, interfaces or their relationships
  - These architectural constraints are recorded as rules based on pre-defined templates
  - The rules verify the integrity of the architecture and report all violations of the architectural constraints

29

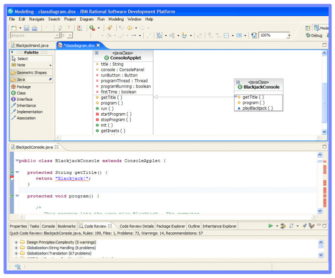
If our goal is to discover defects at early stages, then we can find architectural and design flows before they even reach the coding stage. Rational Software Architects supports a number of architectural and design analysis tools. These tools can examine UML models to discover anti-patterns such as tangles, hubs, and butterflies. In some case, the QuickFix wizards are able to offer fixes for the problem. Out of the box, Rational Software Architect has a rich set of best-practices built into the analysis tools. These best practices can be customized and extended to support team and project design and coding guidelines. This helps drive quality earlier into the development cycle.

Slide 30

Presented by IBM developerWorks

Test web services, Java, J2EE, and Portal applications
IBM Rational Application Developer

- Code Analysis Tools
  - Both static code analysis and dynamic runtime analysis
  - Sequence diagram runtime trace with performance, thread and code coverage
  - Advanced memory leak detection
  - Remote data collection across multiple servers
  - User-defined run-time probes
  - Interactive reports and metrics



```
public class BlackjackConsole extends ConsoleApplet {

   protected String getTitle() {
      return "Blackjack!";
   }

   protected void program() {

      /*
```

Quick Code Review: BlackjackConsole.java, Rules: 199, Files: 1, Problems: 73, Warnings: 14, Recommendations: 57

IBM Rational Application Developer is a comprehensive, integrated development environment, with full support for the J2EE programming model including Web, Java, Web services and EJB development, which accelerates Java development. With integrated Portal development, UML modeling, code analysis, and automated test and deployment tools, Rational Application Developer give Java developers everything they need to be productive and to ensure their code is well-designed, scaleable and ready for production. Built-in version control and team tools enable developers working on complex projects or within large teams coordinate versioning and protect team assets.

Rational Application Developer will assist you in building scaleable Web, Web services (including Web services that participate in SOA) Java, and J2EE or Portal applications. And it will help you:

• Help improve code quality with automated tools for coding standard reviews, component and Web Service unit testing and multi-tier runtime analysis

• Visualize and graphically edit code through the UML Visual Editor for Java and EJB

• Collaborate and share assets across the team using the included Rational ClearCase LT version control

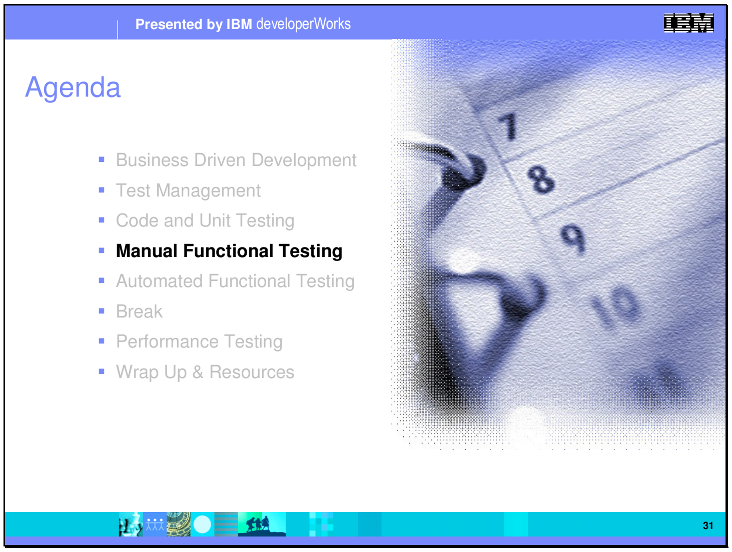
• Adapt and extend your development environment with Eclipse-based plug-ins to match your needs

• Quickly build and deploy interactive reports using drag-and-drop UI components and Crystal Reports software

With these benefits, developers might expect to sacrifice flexibility and the ability to customize their software. Not true here. Rational Application Developer offers a full complement of tools that support traditional code-centric and other low-level controls over middleware artifacts. It also allows you to build, test, debug and deploy your applications without leaving your IDE. And of course, Rational Application Developer is optimized for WebSphere software while supporting multi-vendor runtime environments. For example, in V6 we've added a feature called WebSphere Rapid Deploy to make it easier than ever before to system test and deploy applications to WebSphere Application Server. And for portal development, Rational Application Developer v6 contains enhanced tools for building portlet and portal applications that support both the IBM portlet API as well as the JCP industry standard API.

The end result is that Rational Developer products simplify and accelerate Web, Web services, Java, J2EE, and Portal application development.

Slide 31

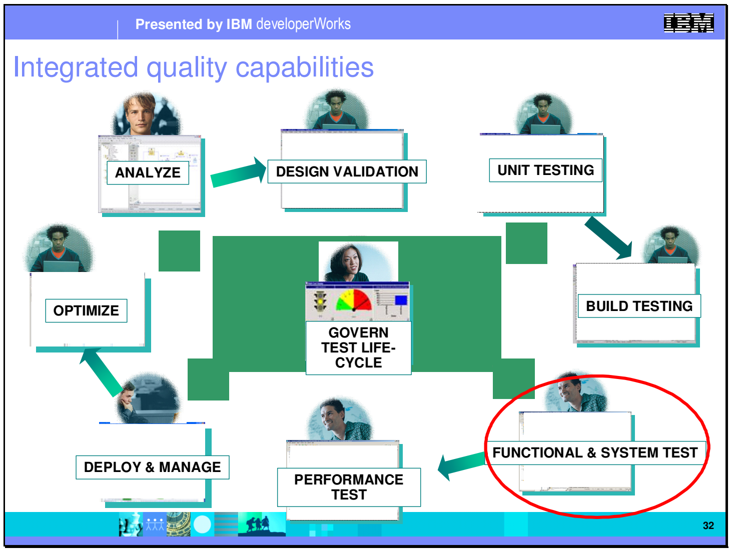


Presented by **IBM** developerWorks

Agenda

- Business Driven Development
- Test Management
- Code and Unit Testing
- **Manual Functional Testing**
- Automated Functional Testing
- Break
- Performance Testing
- Wrap Up & Resources

Slide 32

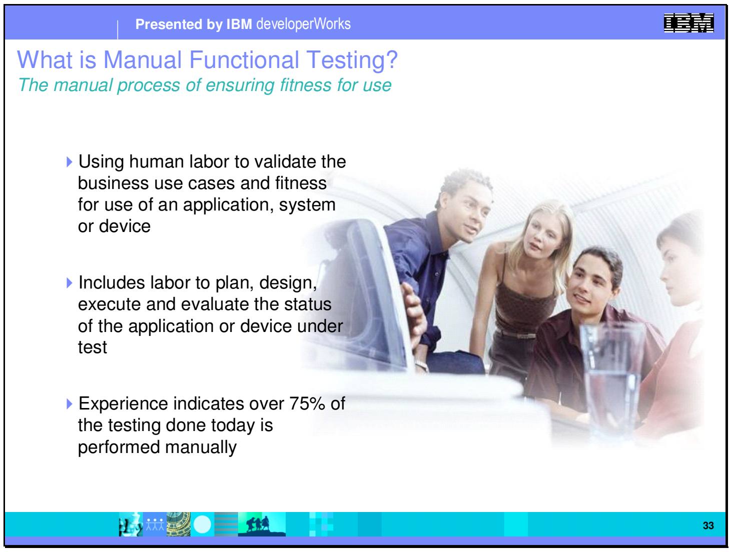


Presented by IBM developerWorks

Integrated quality capabilities

From here we are going to move into the testing stage of development. At this point, builds of the application have been created and made available to the testing team for functional testing. In the next two sections, we are going to break down functional testing into two areas – Manual Functional Testing, testing done manually from written scripts, and Automated Functional Testing, testing done by a tool from prepared scripts and data. We will discuss Manual Functional Testing first.

Slide 33



Presented by IBM developerWorks

What is Manual Functional Testing?

The manual process of ensuring fitness for use

- Using human labor to validate the business use cases and fitness for use of an application, system or device
- Includes labor to plan, design, execute and evaluate the status of the application or device under test
- Experience indicates over 75% of the testing done today is performed manually

33

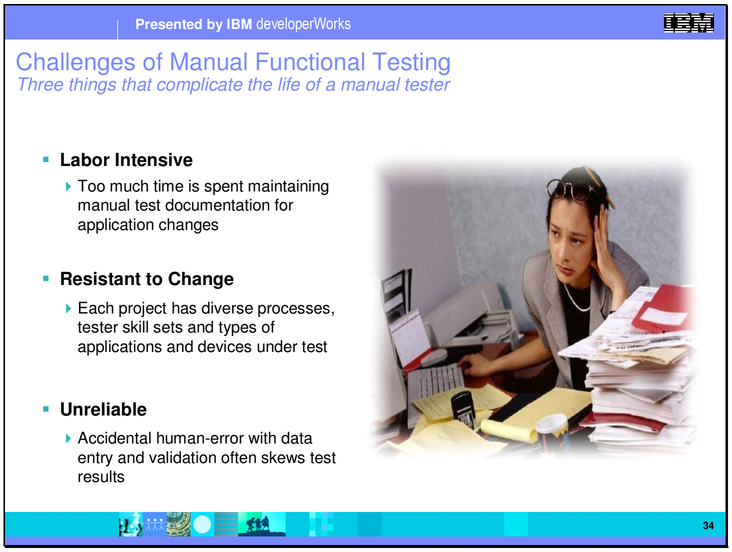
Using human labor to validate the business use cases and fitness for use of an application, system or device

Includes labor to plan, design, execute and evaluate the status of the application or device under test

Experience indicates over 75% of the testing done today is performed manually

In many organizations, tools like Word and Excel are used to document and collate the results of manuals tests

Slide 34



Challenges of Manual Functional Testing

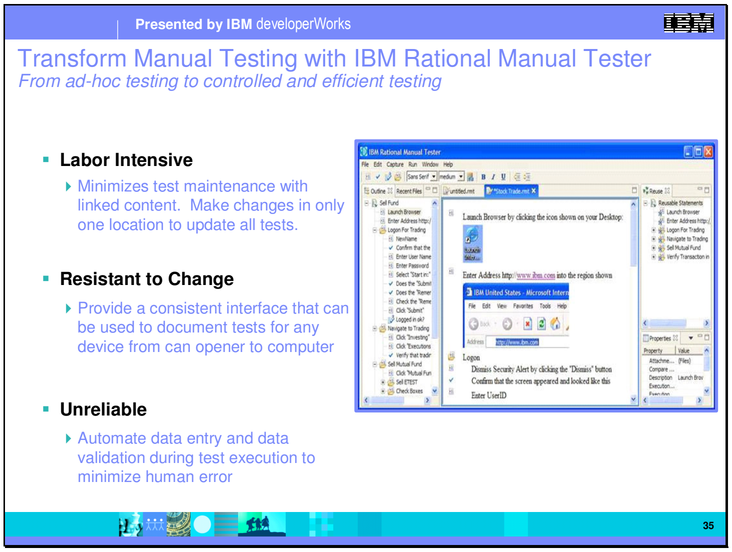
Three things that complicate the life of a manual tester

- **Labor Intensive**
  - Too much time is spent maintaining manual test documentation for application changes
- **Resistant to Change**
  - Each project has diverse processes, tester skill sets and types of applications and devices under test
- **Unreliable**
  - Accidental human-error with data entry and validation often skews test results

Every testing project has different processes, standards to be met or skills sets. This results in reinventing the testing standards for each project

Attempts at automation may have failed due to inability to find a tool that will meet the needs of testing diverse applications, hardware and systems. Further, manual tests can be unreliable due to human error. It is difficult to go through a long manual test script without error, and even harder to reproduce the same test. These are all serious challenges for manual testing.

Slide 35



Presented by IBM developerWorks

Transform Manual Testing with IBM Rational Manual Tester

From ad-hoc testing to controlled and efficient testing

- **Labor Intensive**
  - Minimizes test maintenance with linked content. Make changes in only one location to update all tests.
- **Resistant to Change**
  - Provide a consistent interface that can be used to document tests for any device from can opener to computer
- **Unreliable**
  - Automate data entry and data validation during test execution to minimize human error

Launch Browser by clicking the icon shown on your Desktop:

Enter Address http://www.ibm.com into the region shown

Logon

Dismiss Security Alert by clicking the "Dismiss" button

Confirm that the screen appeared and looked like this

Enter UserID

By dedicating focus on manual testing productivity, Rational Manual Tester helps teams overcome risks to the success of a project. The result is greater focus of time and effort on completing high value, informative manual tests, building a better picture of application quality and greater exposure of functionality requiring improvement before you ship.

IBM Rational Manual Tester is designed to support the diverse skill sets of the people often tasked with manual testing – business analysts, subject-matter experts, and testers

With Rational Manual Tester you can create detailed robust manual tests that are easy for the test executor to follow.

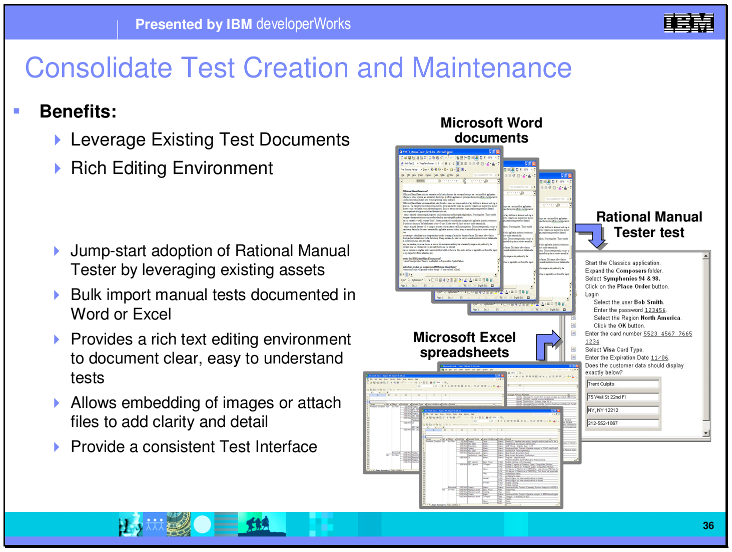
Rational Manual Tester also promotes the usage of linked content, which is the ability to mark test steps as reusable components that can be shared across multiple tests. The advantage is time and cost savings as well as greater control over test script quality and consistency

Rational Manual Tester reduces the likelihood of human area in the areas most prone to mistakes – data entry and data verification

Rational Manual Tester is also designed to support diverse usage models – including support for individuals, teams and mobile workers and diverse processes by allowing customization to reflect a teams internally defined process metrics and values.

Rational Manual Tester is built upon both the Eclipse architectural framework and its test, trace and monitoring extension, named Hyades. Both are open source projects that provide shared, open source services across tooling environments that target application development, testing, deployment and monitoring. This infrastructure provides benefits ranging from the elimination of proprietary datastore formats to enabling support for both internal and third party customization and contribution. An investment in tools based on Eclipse and Hyades avoids vendor lock-in and encourages future innovation

Slide 36



Presented by IBM developerWorks

Consolidate Test Creation and Maintenance

- **Benefits:**
  - Leverage Existing Test Documents
  - Rich Editing Environment
  - Jump-start adoption of Rational Manual Tester by leveraging existing assets
  - Bulk import manual tests documented in Word or Excel
  - Provides a rich text editing environment to document clear, easy to understand tests
  - Allows embedding of images or attach files to add clarity and detail
  - Provide a consistent Test Interface

Microsoft Word documents

Rational Manual Tester test

Start the Classics application.
Expand the **Composers** folder.
Select **Symphonies 94 & 98.**
Click on the **Place Order** button.
Login
Select the user **Bob Smith**.
Enter the password 123456.
Select the Region **North America**.
Click the **OK** button.
Enter the card number 5523 4567 7665 1234
Select **Visa** Card Type.
Enter the Expiration Date 11/06.
Does the customer data should display exactly below?

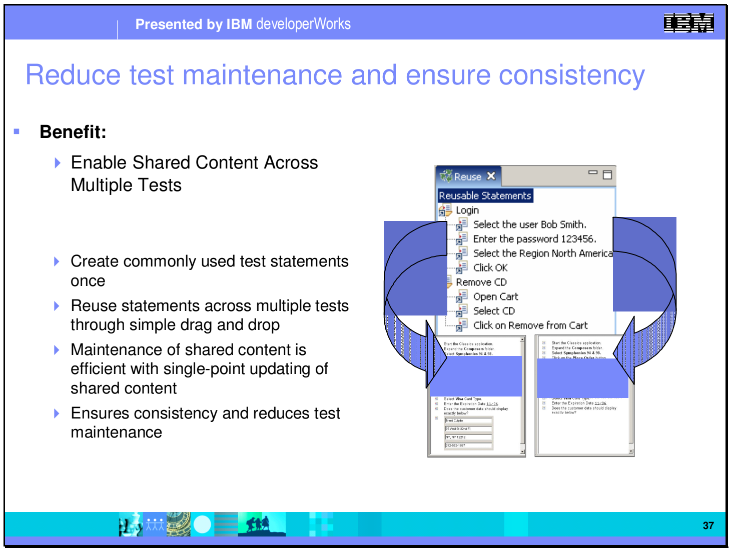
Trent Culpito
75 Wall St 22nd Fl
NY, NY 12212
212-552-1867

Microsoft Excel spreadsheets

RMT allows you to reuse preexisting Word or Excel test documents – either from earlier versions, or from documents created during the analysis phase. It provides the ability to bulk or selectively import information in those documents to jump-start the development of manual tests without have to recreate existing test documentation.

Pictures, validation values, commentary can all be embedded into the manual test document. Additionally, check points where validation should occur can be made part of the test document.

Slide 37



Presented by IBM developerWorks

Reduce test maintenance and ensure consistency

- **Benefit:**
  - Enable Shared Content Across Multiple Tests
  - Create commonly used test statements once
  - Reuse statements across multiple tests through simple drag and drop
  - Maintenance of shared content is efficient with single-point updating of shared content
  - Ensures consistency and reduces test maintenance

Reuse

Reusable Statements

- Login
  - Select the user Bob Smith.
  - Enter the password 123456.
  - Select the Region North America
  - Click OK
- Remove CD
  - Open Cart
  - Select CD
  - Click on Remove from Cart

Start the Classics application.
Expand the **Composers** folder.
Select **Symphonies 94 & 98.**
Select **Visa** Card Type.
Enter the Expiration Date 11/06.
Does the customer data should display exactly below?

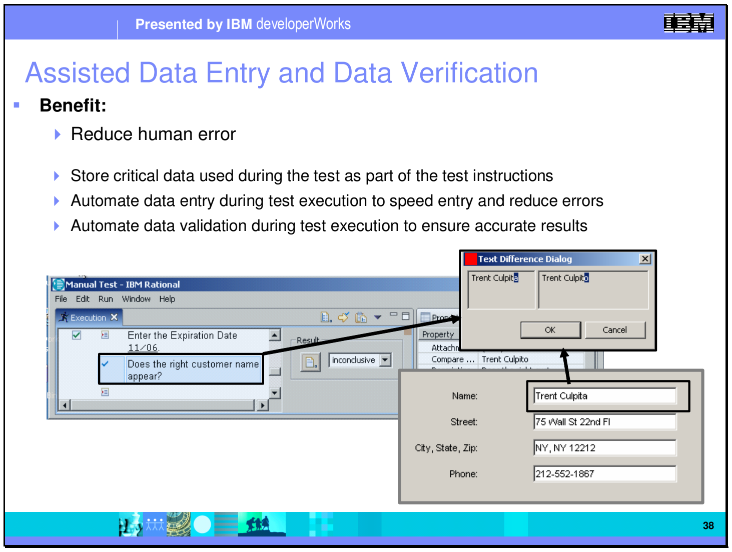
Trent Culpito
75 Wall St 22nd Fl
NY, NY 12212
212-552-1867

Start the Classics application.
Expand the **Composers** folder.
Select **Symphonies 94 & 98.**
Click on the **Place Order** button.
Select **Visa** Card Type.
Enter the Expiration Date 11/06.
Does the customer data should display exactly below?

37

Rational Manual Tester promotes the usage of linked content – the ability to define particular test step blocks as reusable components that can be shared across multiple tests. The advantage is not simply reuse, a time and cost saving measure under any circumstance. The advantage is also greater control over test script quality and consistency. Modification of linked content is automatically reflected in all test scripts using that content, ensuring change is communicated accurately to the entire test team.

Slide 38



Presented by IBM developerWorks

Assisted Data Entry and Data Verification

- **Benefit:**
  - Reduce human error
  - Store critical data used during the test as part of the test instructions
  - Automate data entry during test execution to speed entry and reduce errors
  - Automate data validation during test execution to ensure accurate results

☑ Enter the Expiration Date 11/06.

✓ Does the right customer name appear?

inconclusive

Compare ... Trent Culpito

Trent Culpita | Trent Culpito

Name: Trent Culpita

Street: 75 Wall St 22nd Fl

City, State, Zip: NY, NY 12212

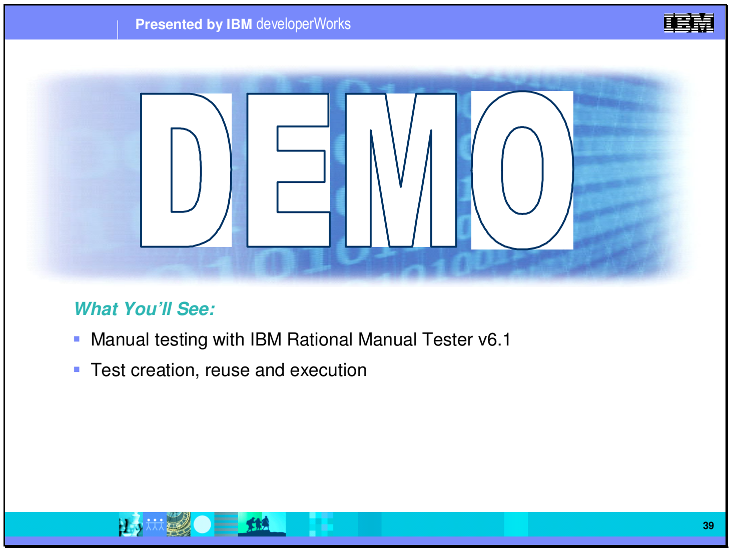
Phone: 212-552-1867

38

Second, data copied from an application user interface is automatically compared with expected values, automating the comparison and storing the result in a consolidated test report.

These capabilities ensure that data is entered accurately and quickly and that data validation of large amounts of data can be done without introducing the possibility of human error.

Slide 39

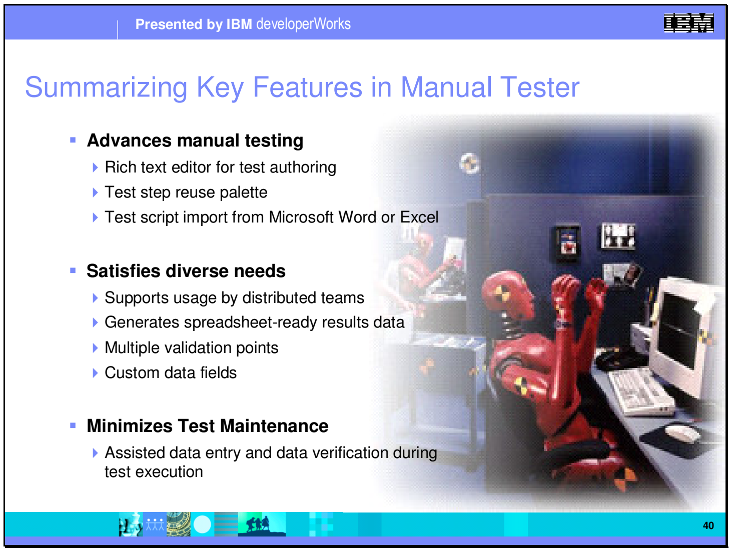


DEMO

What You'll See:

- Manual testing with IBM Rational Manual Tester v6.1
- Test creation, reuse and execution

Slide 40



Presented by IBM developerWorks

Summarizing Key Features in Manual Tester

- **Advances manual testing**
  - Rich text editor for test authoring
  - Test step reuse palette
  - Test script import from Microsoft Word or Excel
- **Satisfies diverse needs**
  - Supports usage by distributed teams
  - Generates spreadsheet-ready results data
  - Multiple validation points
  - Custom data fields
- **Minimizes Test Maintenance**
  - Assisted data entry and data verification during test execution

With Rational Manual Tester, we can move beyond just the use of Word documents or Excel spreadsheets. With a rich text editor, we can import existing tests, and get reuse of tests by linking to test modules.

This satisfies the diverse needs in testing across multiple different projects. It supports the needs of distributed teams, and generates test results in spreadsheet ready results data. With the assisted data entry feature, we can eliminate much of the sources of human error by doing automated results verification.

Slide 41

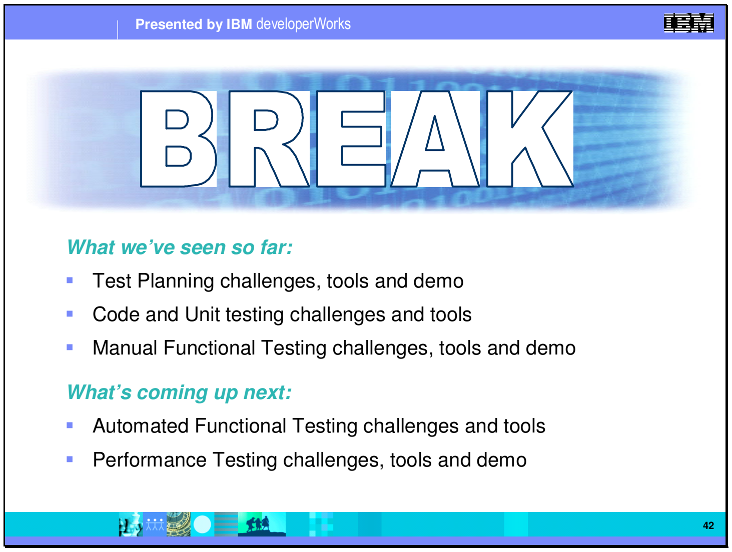
Presented by IBM developerWorks

Agenda

- Business Driven Development
- Test Management
- Code and Unit Testing
- Manual Functional Testing
- **Break**
- Automated Functional Testing
- Performance Testing
- Wrap Up & Resources

.

Slide 42



BREAK

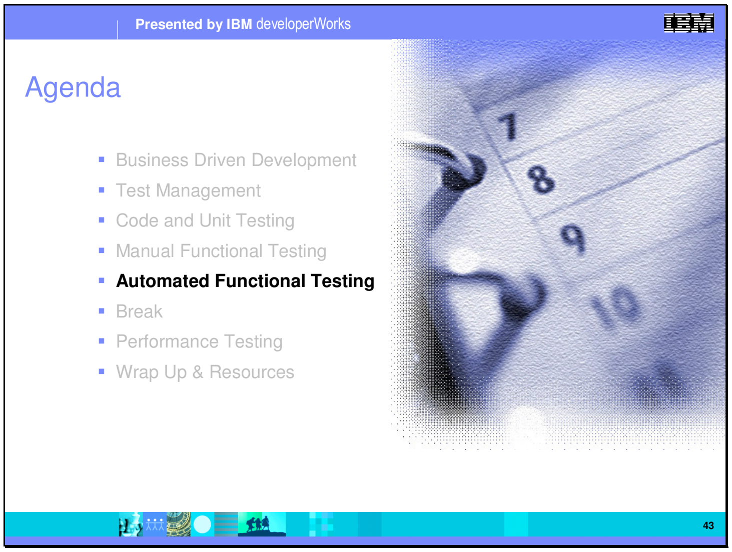
What we've seen so far:

- Test Planning challenges, tools and demo
- Code and Unit testing challenges and tools
- Manual Functional Testing challenges, tools and demo

What's coming up next:

- Automated Functional Testing challenges and tools
- Performance Testing challenges, tools and demo

Slide 43

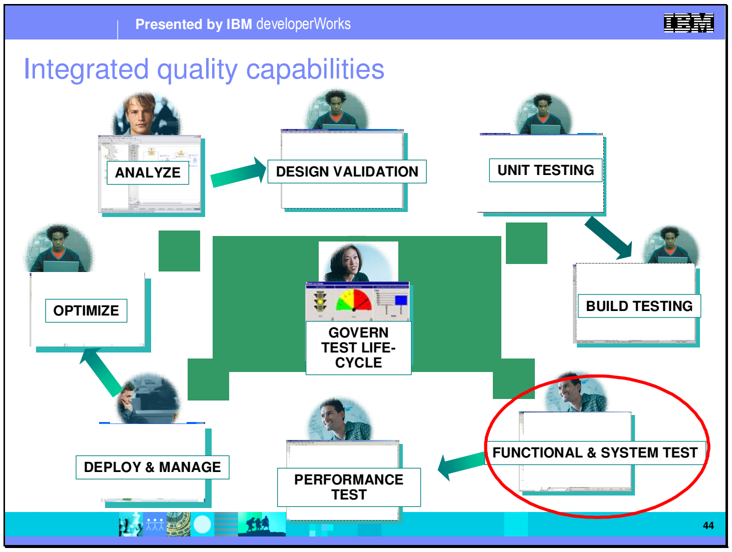


Agenda

- Business Driven Development
- Test Management
- Code and Unit Testing
- Manual Functional Testing
- **Automated Functional Testing**
- Break
- Performance Testing
- Wrap Up & Resources

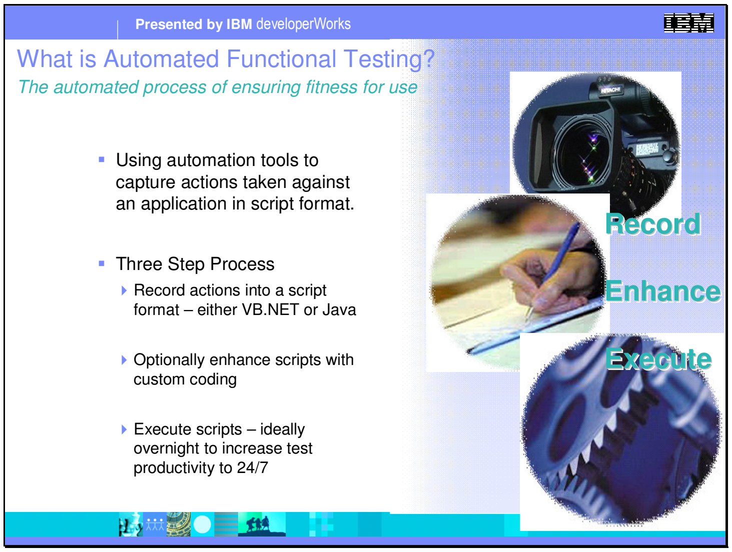
.

Slide 44



In this section, we will move on to Automated Functional testing. It is similar in concept to manual testing, but the testing functions are done without human intervention, and are controlled by test scripts and test data pools.

Slide 45



Presented by IBM developerWorks

What is Automated Functional Testing?

The automated process of ensuring fitness for use

- Using automation tools to capture actions taken against an application in script format.
- Three Step Process
  - Record actions into a script format – either VB.NET or Java
  - Optionally enhance scripts with custom coding
  - Execute scripts – ideally overnight to increase test productivity to 24/7

Record

Enhance

Execute

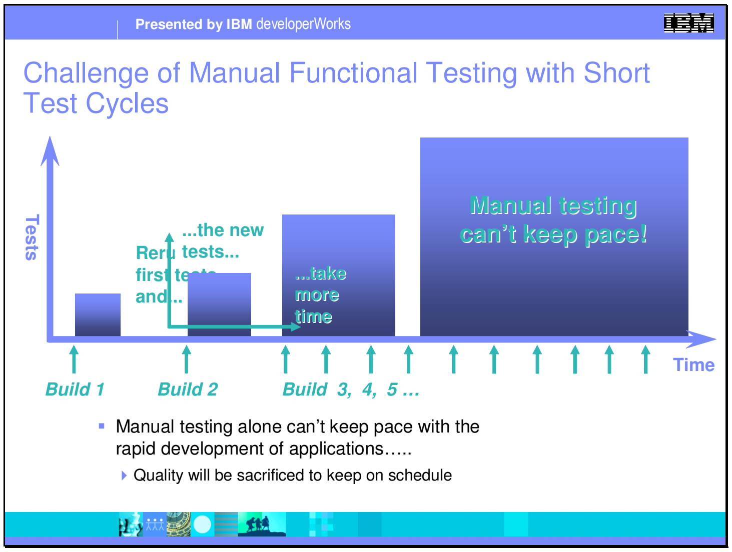
Automated functional testing using automation tools to take action based on a script format. Basically this means capturing the user interactions on an application such as field entries. This gets captured into a script language which can be played back at a later time duplicating the user interactions. It's a three step process.

You create the script typically buy capturing a user session. The scripting language can be either VB.NET of Java.

You can optionally enhance the script by directly editing it. About 80% of the time, the captured session is sufficient for testing, but there are instances where iterating over sections with different data values, or other variations can enhance the test.

Execute the scripts. Since this can be a lengthy process, doing them overnight is a good idea. You collect the report the next day.

Slide 46



Presented by IBM developerWorks

Challenge of Manual Functional Testing with Short Test Cycles

- Manual testing alone can't keep pace with the rapid development of applications…..
  - Quality will be sacrificed to keep on schedule

Before we go on to discuss an automation use case, let's discuss first why we need automation.

This graph depicts the relationship between the number of tests and the time it takes to run them. In a manual environment, if you run all your tests on build 1 it takes a certain amount of time.

Now, in build 2, you need to rerun all the tests you did in build 1, and that takes the same amount of time it did before, and...

...the new tests you've added for the new functionality in Build 2 require even more time. It's easy to see where the continuation of this process eventually takes us....

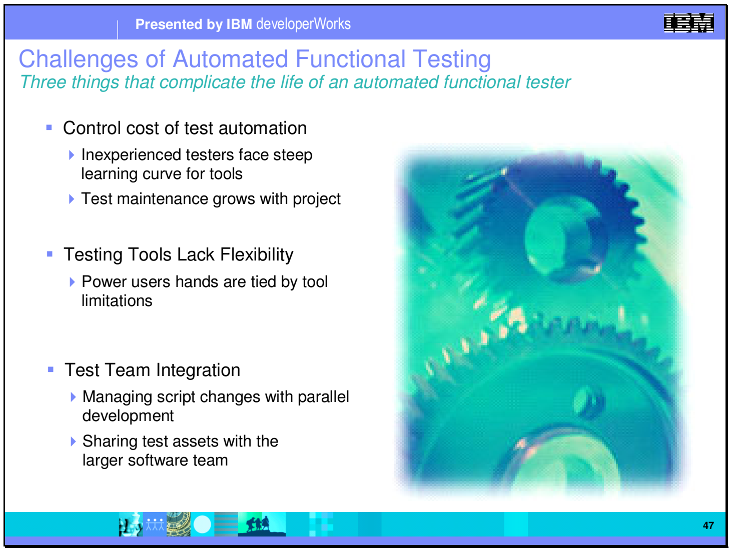
Manual testing can't keep pace with an iterative environment.

The difficulty lies in the amount of time each test cycle takes vs. the speed at which Development can deliver new builds. In this example, the QA group starts a test cycle on build 3, but before they are finished Development delivers build #4...and 5 and 6... and so on.

Ironically, the problem arises from the fact that iterative development works!

Modern GUI development technology is very advanced. It offers the developer rapid development power. This means they can change the application very quickly and deliver new versions to testing. In fact, many companies deliver a new build <u>every day.</u> It is physically impossible for any number of people manually testing to keep up with this rapid development.

Slide 47



Presented by IBM developerWorks

Challenges of Automated Functional Testing

Three things that complicate the life of an automated functional tester

- Control cost of test automation
  - Inexperienced testers face steep learning curve for tools
  - Test maintenance grows with project
- Testing Tools Lack Flexibility
  - Power users hands are tied by tool limitations
- Test Team Integration
  - Managing script changes with parallel development
  - Sharing test assets with the larger software team

47

There are three challenges with using automated testing:

Controlling the cost of test automation

Automated testing tools can have steep learning curves. There is more investment in learning an automated tool then there is in running a manual script. Also, automated test scripts need to be maintained as the project grows. From build to build, we will be getting more functionality, which means more scripts, and these scripts need to be maintained. Developers look at visual changes in a user interface – moving a button, changing a label or color – as inconsequential, but to a testing script, they are not. All this takes tie and money

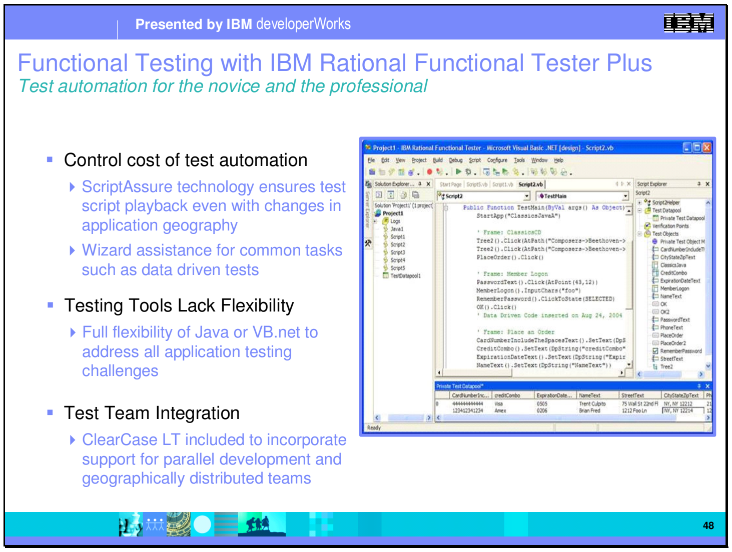
Lack of flexibility in testing tools

Testing tools in the past have lacked flexibility. This “tied the hands” of power users that wanted more flexibility from the scripting language.

Integration of test teams with development

With rapid development and iterative builds, there needs to be a tighter linkage between the development teams and the testing teams. We want to be able to share test assets with the development team – test scripts, test results, and defects. We also need to manage script changes with parallel development. The testing team needs to be able to be working on multiple versions of a testing script for the different builds of the application.

Slide 48



Presented by IBM developerWorks

Functional Testing with IBM Rational Functional Tester Plus

Test automation for the novice and the professional

- Control cost of test automation
  - ScriptAssure technology ensures test script playback even with changes in application geography
  - Wizard assistance for common tasks such as data driven tests
- Testing Tools Lack Flexibility
  - Full flexibility of Java or VB.net to address all application testing challenges
- Test Team Integration
  - ClearCase LT included to incorporate support for parallel development and geographically distributed teams

Project1 - IBM Rational Functional Tester - Microsoft Visual Basic .NET [design] - Script2.vb
File Edit View Project Build Debug Script Configure Tools Window Help
Solution Explorer
Start Page | Script5.vb | Script1.vb | Script2.vb
Script2
TestMain
Script Explorer
Private Test Datapool*
Ready

```
Public Function TestMain(ByVal args() As Object)
    StartApp("ClassicsJavaA")

    ' Frame: ClassicsCD
    Tree2().Click(AtPath("Composers->Beethoven->
    Tree2().Click(AtPath("Composers->Beethoven->
    PlaceOrder().Click()

    ' Frame: Member Logon
    PasswordText().Click(AtPoint(43,12))
    MemberLogon().InputChars("foo")
    RememberPassword().ClickToState(SELECTED)
    OK().Click()
    ' Data Driven Code inserted on Aug 24, 2004

    ' Frame: Place an Order
    CardNumberIncludeTheSpacesText().SetText(DpS
    CreditCombo().SetText(DpString("creditCombo"
    ExpirationDateText().SetText(DpString("Expir
    NameText().SetText(DpString("NameText"))
```

	CardNumberInc...	creditCombo	ExpirationDate...	NameText	StreetText	CityStateZipText
0	444444444444	Visa	0505	Trent Culpito	75 Wall St 22nd Fl	NY, NY 12212
	123412341234	Amex	0206	Brian Fred	1212 Foo Ln	NY, NY 12214

The Rational Functional Tester addresses these issues effectively. Looking at the same three issues:

Controlling the cost of test automation

Inside Rational Functional Tester is a technology we call Script Assure. It helps with the maintenance of test scripts from one build to another by being able to ignore superficial changes to user interfaces.

To help accelerate the productivity of new testers, there are wizards that assist in the common tasks done in script production such as data driven tests. This allows inexperienced testers to become productive quickly.

Lack of flexibility in testing tools

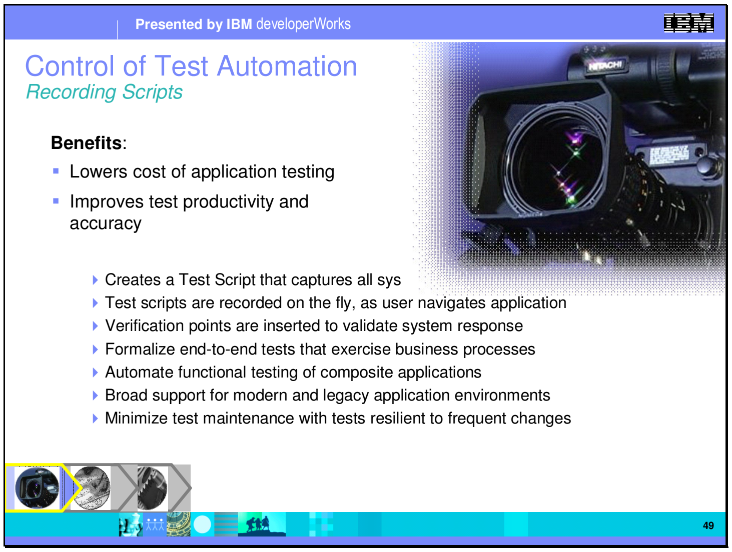
For experienced testers, they have the full power of the scripting language to work with – either Java, or VB .net. Both scripting languages are widely used, and powerful tools for building complex scripts.

Integration of test teams with development

The Rational Functional Tester is integrated with the development teams in two ways. The tool itself is integrated into the same framework as the development teams – Eclipse for Java development, and VB .Net for .Net development. Second, use of a configuration management tool such as Rational ClearCase allows the test assets to be associated with the right build versions. Rational Functional Tester comes with a license for ClearCase LT, a good solution for small teams – local, or geographically distributed.

Enhancements and defects can be controlled with Rational ClearQuest, and associated with the correct build. This further helps in the integration of the two test and development teams and mitigates many of the problems associated with rapid development.

Slide 49



Presented by IBM developerWorks

Control of Test Automation

Recording Scripts

Benefits:

- Lowers cost of application testing
- Improves test productivity and accuracy
    - Creates a Test Script that captures all sys
    - Test scripts are recorded on the fly, as user navigates application
    - Verification points are inserted to validate system response
    - Formalize end-to-end tests that exercise business processes
    - Automate functional testing of composite applications
    - Broad support for modern and legacy application environments
    - Minimize test maintenance with tests resilient to frequent changes

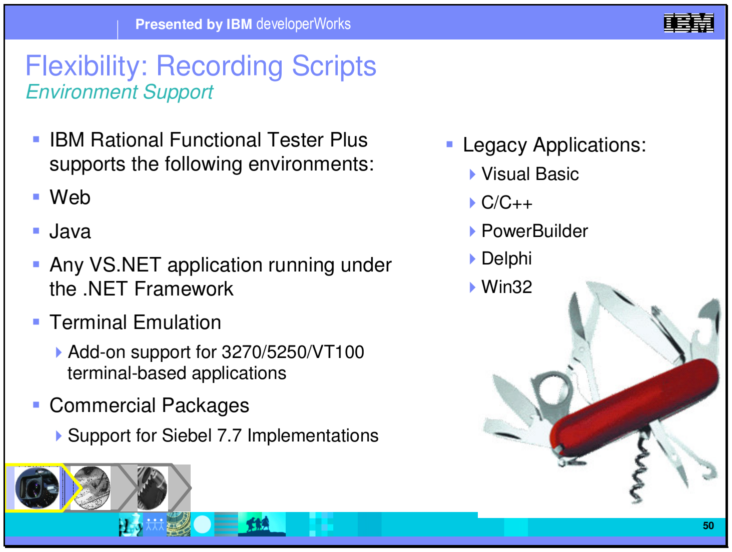
We’ll now walk through each of those three stages in detail and examine features relevant to each stage and how they support the challenges in the first section.

With scripts that are recorded, we can play them back at any time and do the same level of testing. This helps to lower the cost of application testing, because we now have a reusable asset. Further, this asset becomes one of the artifacts with a particular build so we can always go back to previous versions and have all of the tests for that version available.

Because the scripts are created on the fly as the tester navigates through the application functionality. This means as you go along you can insert data driven scripts, or script verification points with out having to stop.   You don’t have to go back and add those in later.

Broad support for both modern and legacy application environments and resiliency to changes further reduces the costs of doing testing.

Slide 50



Flexibility: Recording Scripts

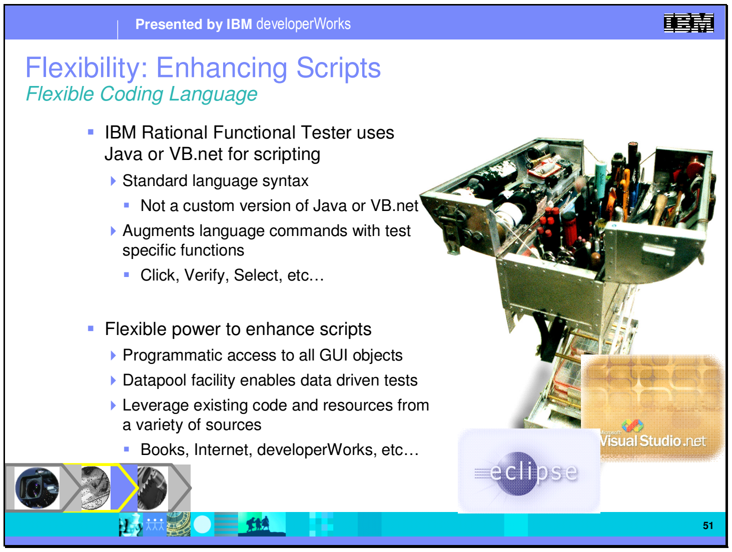
Environment Support

- IBM Rational Functional Tester Plus supports the following environments:
- Web
- Java
- Any VS.NET application running under the .NET Framework
- Terminal Emulation
  - Add-on support for 3270/5250/VT100 terminal-based applications
- Commercial Packages
  - Support for Siebel 7.7 Implementations

- Legacy Applications:
  - Visual Basic
  - C/C++
  - PowerBuilder
  - Delphi
  - Win32

This picture shows the range of environments and languages in those environments that are supported by the Rational Functional Tester. In future releases these environments will be expanded.

Slide 51



Presented by IBM developerWorks

Flexibility: Enhancing Scripts

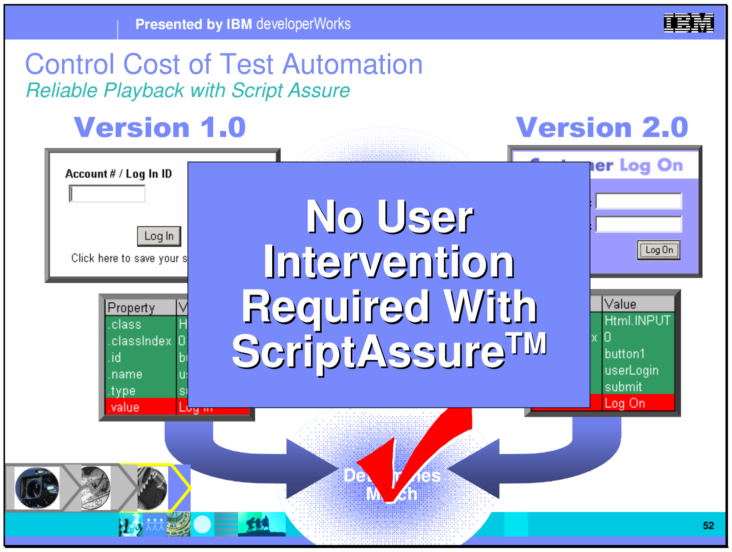
Flexible Coding Language

- IBM Rational Functional Tester uses Java or VB.net for scripting
  - Standard language syntax
    - Not a custom version of Java or VB.net
  - Augments language commands with test specific functions
    - Click, Verify, Select, etc…
- Flexible power to enhance scripts
  - Programmatic access to all GUI objects
  - Datapool facility enables data driven tests
  - Leverage existing code and resources from a variety of sources
    - Books, Internet, developerWorks, etc…

The Functional Tester uses Java or VB.net as the scripting language. When you install the product, you can choose to use a Eclipse shell and use Java scripting, use Visual Studio and VB.net. The benefits of both are that you can use a standard scripting language to augment the scripts that are produced during recording. With a standard scripting language, you can leverage existing assets from a wide range of sources; books, internet groups, and developerWorks.

In addition, within the scripts, you have full access to the GUI objects, and can use the datapool facility to create data driven tests.

Slide 52



Presented by IBM developerWorks

Control Cost of Test Automation

Reliable Playback with Script Assure

Version 1.0

Version 2.0

No User Intervention Required With ScriptAssure™

Let’s take a closer look at the ScriptAssure technology. ScriptAssure is the technology that makes test scripts immune to object name changes between software builds. This is a critical feature to reduce script maintenance and increase script reliability.

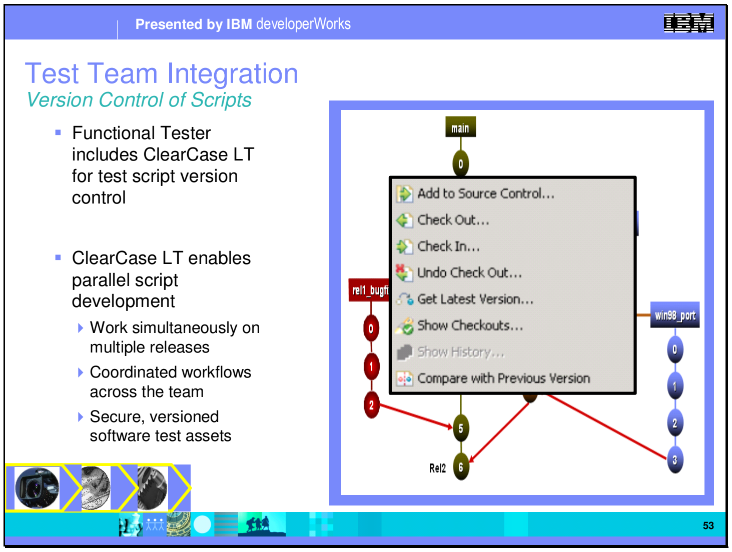
Here we have two builds of the same application.

The scripting tool sees the underlying attributes which allow the tool to uniquely identify the objects being testing. The problem is, when the attributes are changed build to build as the application gets constructed, the test scripts can no longer uniquely identify the object to be tested, thus breaks. In the past, the Tester would need to manually intervene using a GUI Mapper to identify to the script what it is was supposed to test.

Unique to Rational, the new technology tool can identify objects even after the identifiers have changed. It uses many different attributes, weighs them proportionately, then uses a sensitivity setting to identify the object. This way, should one or a few properties – such as an object name – change between builds, it can still recognize the correct object.

The benefit is that even as object properties change between builds, the scripts will still playback. This decreases script failures by making scripts more resilient to change. Of course, all of this translates to less script maintenance and more time for the Tester.

Slide 53



Presented by IBM developerWorks

Test Team Integration

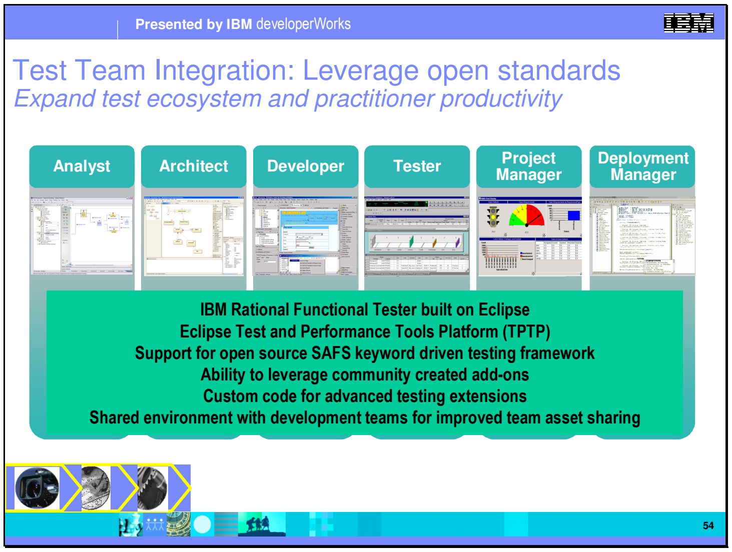
Version Control of Scripts

- Functional Tester includes ClearCase LT for test script version control
- ClearCase LT enables parallel script development
  - Work simultaneously on multiple releases
  - Coordinated workflows across the team
  - Secure, versioned software test assets

One aspect of team integration is through the Configuration Management system. All of the Rational testing products ship with a license for ClearCase LT.

ClearCase LT enables test script version control. This means that test scripts that are associated with a particular build or version of a product can be stored independent of other versions. It allows parallel script development across geographically dispersed development teams, and can easily be scaled to support replicated multiple sites. The benefits are that work can happen simultaneously on multiple releases, and the testing workflows can be coordinated across the whole team, and that the software test assets are maintained in a secure versioned form.

Slide 54



Presented by IBM developerWorks

Test Team Integration: Leverage open standards

Expand test ecosystem and practitioner productivity

IBM Rational Functional Tester built on Eclipse
Eclipse Test and Performance Tools Platform (TPTP)
Support for open source SAFS keyword driven testing framework
Ability to leverage community created add-ons
Custom code for advanced testing extensions
Shared environment with development teams for improved team asset sharing

Rational’s software quality tools are committed to the support of open and standard development environments and languages in order to ensure easy sharing of data among team members and tools, tight integration to the practitioner’s natural environments and the ability to leverage the skill sets that are commonly used among software development team members.

RFT integrates into either Eclipse or VS.NET IDE

RFT supports both Java and VB.net scripting languages so practitioners can use the language they are most familiar with

TPTP is an open source Eclipse project with a standard format for storing and sharing test and performance data. Use of this format allows us to share data among practitioner development and testing tools and Tivoli application monitoring tools

API’s for test management integration allow the inclusion of 3rd party test tools in our test management ecosystem.

Visit our **Software Automation Framework Support** SAFSDEV project on SourceForge.

Welcome to the SAFSDEV project!
Our focus is on implementing tools and frameworks for strategic, long-term success in functional test automation. To that end we work on "**keyword-driven**" engines, also sometimes called "data-driven" engines. What we are dealing with, actually, are tools and full implementations of testing frameworks for "**action-based testing**" or "the action word approach" for functional test automation. Read more about the benefits of keyword-driven testing in the Test Automation Frameworks whitepaper.

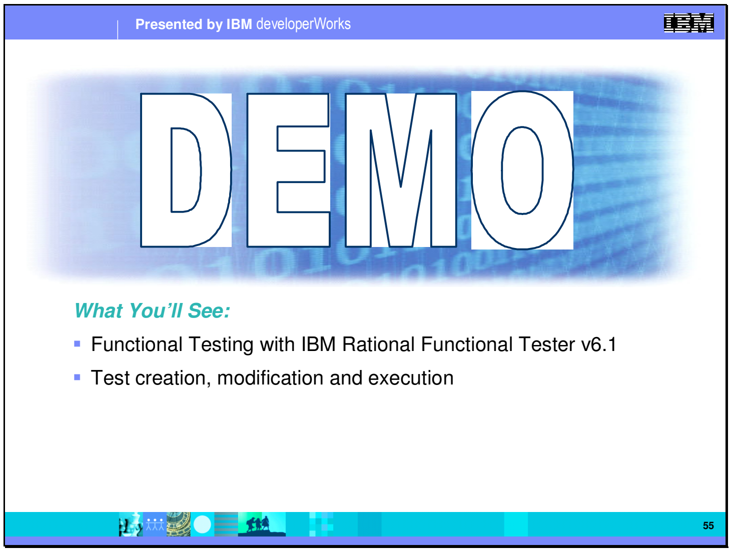
Larger portions of our tools development are intended to benefit functional test automation in general. This is an effort to provide functionality and tools that augment, or can be used independent of, the commercial automation solutions used by many automators. These tools are not only suitable for the keyword-driven engines that are the focal point of our effort, but they are also useful for any number of other software test automation projects.

We have dubbed the "general-purpose" nature of our work, **Software Automation Framework Support**, or **SAFS**. We then take these general-purpose tools and apply them to specific keyword-driven engine implementations we call, **SAFS Engines.** Our primary goal is to implement a common keyword-driven engine design across multiple tools and testing platforms for robust automation that can be immune to the course of any one particular testing tool. This helps us realize the concept:

"We must minimize the impact caused by changes in the applications we are testing, and changes in the tools we use to test them."

This strategy allows us to separate our tests from the automation tools that will execute them. So, we can migrate tests from one automation tool to another, or even use multiple automation tools during a single test. For example, we could use IBM Rational Robot, Abbot, and IBM Rational Functional Tester at the same time to accomplish our goals. We could also have IBM Rational Robot execute the test on Windows, but use Abbot to run the same test on Unix.

Slide 55

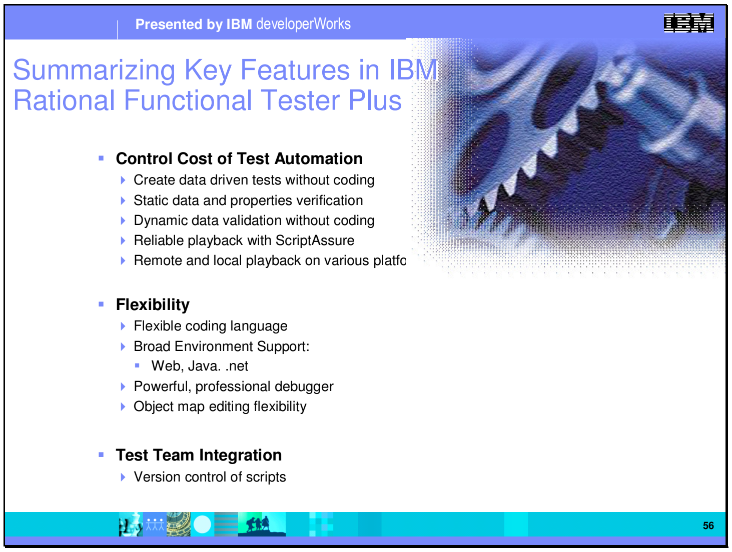


DEMO

What You'll See:

- Functional Testing with IBM Rational Functional Tester v6.1
- Test creation, modification and execution

Slide 56



Presented by IBM developerWorks

IBM

Summarizing Key Features in IBM Rational Functional Tester Plus

- **Control Cost of Test Automation**
  - Create data driven tests without coding
  - Static data and properties verification
  - Dynamic data validation without coding
  - Reliable playback with ScriptAssure
  - Remote and local playback on various platfc
- **Flexibility**
  - Flexible coding language
  - Broad Environment Support:
    - Web, Java. .net
  - Powerful, professional debugger
  - Object map editing flexibility
- **Test Team Integration**
  - Version control of scripts

We believe that Rational Functional Tester has some key benefits which address automated testing issues. The three key areas are;

Cost Control

The Functional Tester helps control test costs in two primary ways. It helps tester productivity by giving the ability to create data driven tests without any coding. It also helps extend the life of automated tests over successive builds by making them more robust with Script Assure.

Flexibility

Flexibility is achieved by the use of standard well know scripting languages – Java Script, and VB.net.

Team Integration

Team integration is achieved with use of the included ClearCase LT configuration management system providing version control of test scripts, and through the integration of the testing tool on top of the same development IDE – either Eclipse, or VisualStudio.net based.

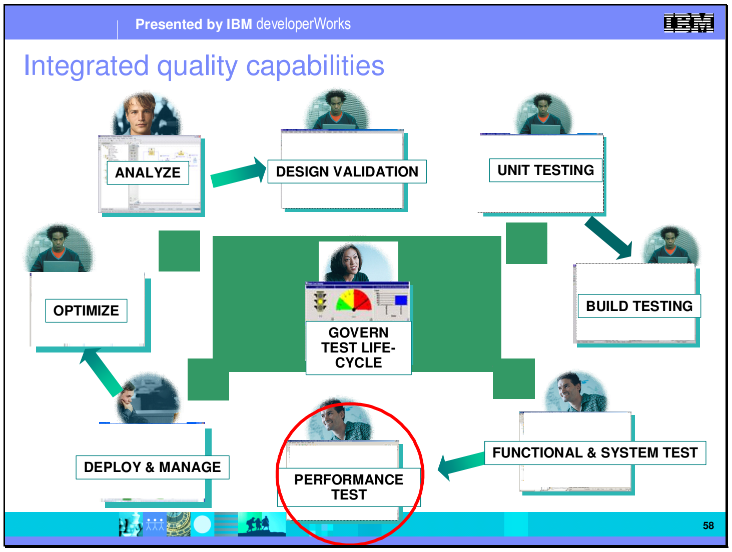
Slide 57

Agenda

- Business Driven Development
- Test Management
- Code and Unit Testing
- Manual Functional Testing
- Automated Functional Testing
- Break
- **Performance Testing**
- Wrap Up & Resources

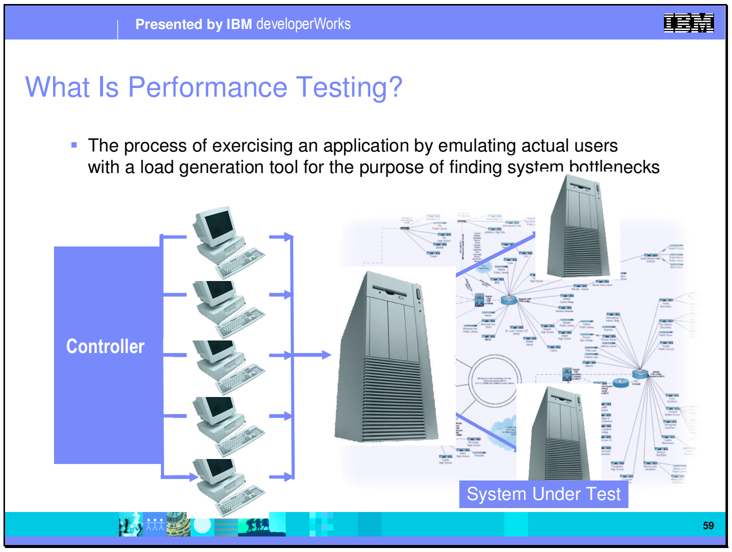
.

Slide 58



After functional testing has been completed, testing the product for performance and scalability is the next step.

Slide 59



Presented by **IBM** developerWorks

What Is Performance Testing?

- The process of exercising an application by emulating actual users with a load generation tool for the purpose of finding system bottlenecks

59

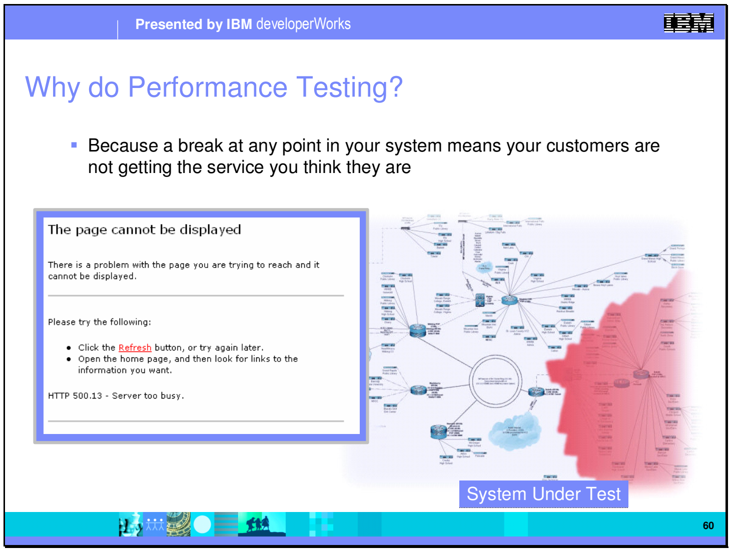
First, what is performance testing?

It is the process of exercising an application by emulating actual users putting a load on your system for the purpose of finding system bottlenecks. This load is generally created by a load generation tool. You typically have a master computer that is controlling other computers – as well as ‘virtual testers’ – to emulate this production load.

Once the test run is complete all the data is returned to the master where the user analyzes the data to find the performance bottlenecks.

There are probably a million different definitions which we could use, but I’m hoping we can just start from here and build on this. From here we’ll move forward take a look at the various challenges of performance testing.

Slide 60



Why do Performance Testing?

- Because a break at any point in your system means your customers are not getting the service you think they are

The page cannot be displayed

There is a problem with the page you are trying to reach and it cannot be displayed.

Please try the following:

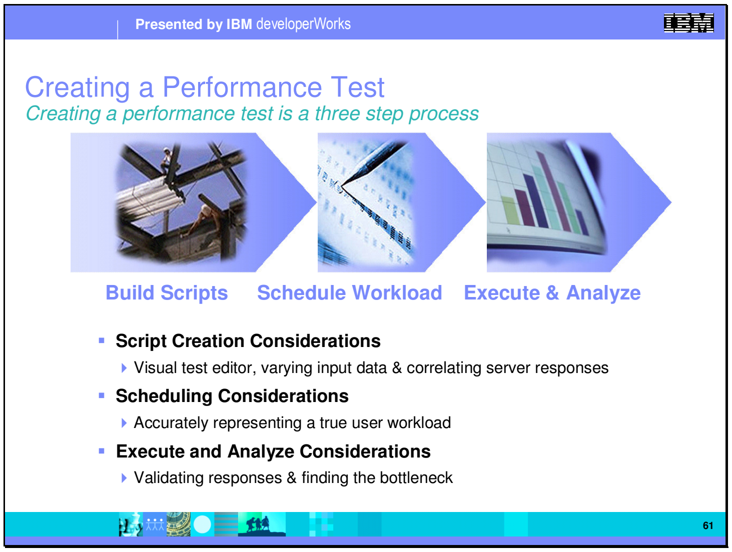
- Click the Refresh button, or try again later.
- Open the home page, and then look for links to the information you want.

HTTP 500.13 - Server too busy.

The question is not so much “Why should you do performance testing?”, it is more “What is the cost of not doing performance testing?” Enterprise websites are one of the primary sales channels for many companies. If a transaction is perceptibly slow to the user, or times out and is lost because of server performance, there is a good chance of loosing that sale, and possibly the customer. It is not sufficient to functionally test at just normal loads, you need to test simulating the unusual loads.

Just as we want to find functional errors before the customer finds them, we want to find performance “errors” as well.

Slide 61



Presented by IBM developerWorks

Creating a Performance Test

Creating a performance test is a three step process

Build Scripts Schedule Workload Execute & Analyze

- **Script Creation Considerations**
  - Visual test editor, varying input data & correlating server responses
- **Scheduling Considerations**
  - Accurately representing a true user workload
- **Execute and Analyze Considerations**
  - Validating responses & finding the bottleneck

61

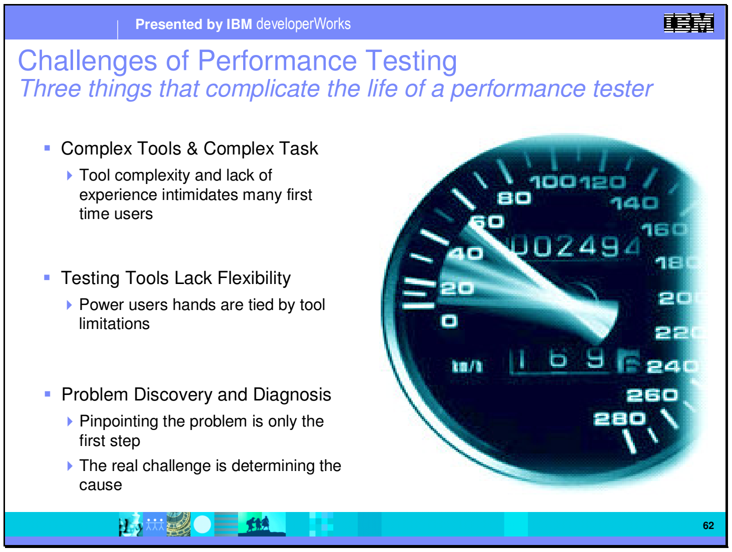
In performance testing, the same basic steps are much the same as in functional testing.

Script Creation. Here we are recording a script. The difference here is that we are capturing the traffic going back and forth between the client and the server. What that does is enable us to scale up that traffic easily and quickly. Using the visual test editor, we can vary the input data to simulate different users.

Scheduling Considerations. Here we want to simulate different usage patterns. We can simulate normal expected types of usage, and unusual types of usage. This will give us the information about the full range of usage patterns.

Execute and Analyze. Here we want to validate the results and responses, and analyze the system to find performance bottlenecks. One of the Performance Tester’s key strengths is analysis, helping us to quickly identify performance bottlenecks.

Slide 62



Challenges of Performance Testing

Three things that complicate the life of a performance tester

- Complex Tools & Complex Task
  - Tool complexity and lack of experience intimidates many first time users
- Testing Tools Lack Flexibility
  - Power users hands are tied by tool limitations
- Problem Discovery and Diagnosis
  - Pinpointing the problem is only the first step
  - The real challenge is determining the cause

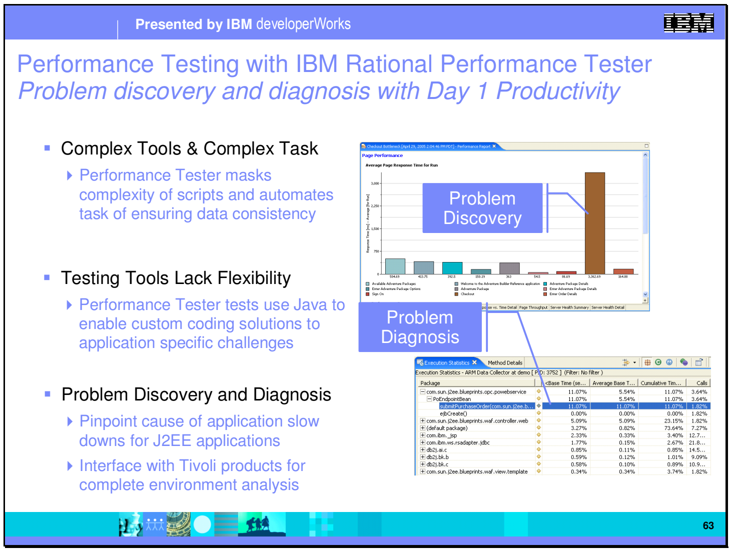
Some of the challenges of performance testing. Like the functional testing, there are three issues.

Tool Complexity. Tool complexities, along with lack of knowledge of performance testing in general are often barriers to doing performance testing.

Flexibility. Power users are often limited in what the can do with a tool – tying their hands so to speak.

Discovery and Diagnosis. The final challenge is determining the root cause of the performance bottleneck from all the data that is collected.

Slide 63


Checkout Bottleneck [April 29, 2005 2:04:46 PM PDT] - Performance Report
Page Performance
Average Page Response Time for Run
Response Time [ms] -- Average [for Run]
3,000
2,250
1,500
750
0
504.69
413.75
392.5
150.19
363
54.5
86.69
3,362.69
164.88
Available Adventure Packages
Enter Adventure Package Options
Sign On
Welcome to the Adventure Builder Reference application
Adventure Package
Checkout
Adventure Package Details
Enter Adventure Package Details
Enter Order Details
Response vs. Time Detail
Page Throughput
Server Health Summary
Server Health Detail
Problem Discovery
Problem Diagnosis
Execution Statistics
Method Details
63

Presented by IBM developerWorks

Performance Testing with IBM Rational Performance Tester

Problem discovery and diagnosis with Day 1 Productivity

- Complex Tools & Complex Task
  - Performance Tester masks complexity of scripts and automates task of ensuring data consistency
- Testing Tools Lack Flexibility
  - Performance Tester tests use Java to enable custom coding solutions to application specific challenges
- Problem Discovery and Diagnosis
  - Pinpoint cause of application slow downs for J2EE applications
  - Interface with Tivoli products for complete environment analysis

Execution Statistics - ARM Data Collector at demo [PID: 3752] (Filter: No filter)

Package	<Base Time (se...	Average Base T...	Cumulative Tim...	Calls
com.sun.j2ee.blueprints.opc.powebservice	11.07%	5.54%	11.07%	3.64%
PoEndpointBean	11.07%	5.54%	11.07%	3.64%
submitPurchaseOrder(com.sun.j2ee.b...	11.07%	11.07%	11.07%	1.82%
ejbCreate()	0.00%	0.00%	0.00%	1.82%
com.sun.j2ee.blueprints.waf.controller.web	5.09%	5.09%	23.15%	1.82%
(default package)	3.27%	0.82%	73.64%	7.27%
com.ibm._jsp	2.33%	0.33%	3.40%	12.7...
com.ibm.ws.rsadapter.jdbc	1.77%	0.15%	2.67%	21.8...
db2j.ai.c	0.85%	0.11%	0.85%	14.5...
db2j.bk.b	0.59%	0.12%	1.01%	9.09%
db2j.bk.c	0.58%	0.10%	0.89%	10.9...
com.sun.j2ee.blueprints.waf.view.template	0.34%	0.34%	3.74%	1.82%

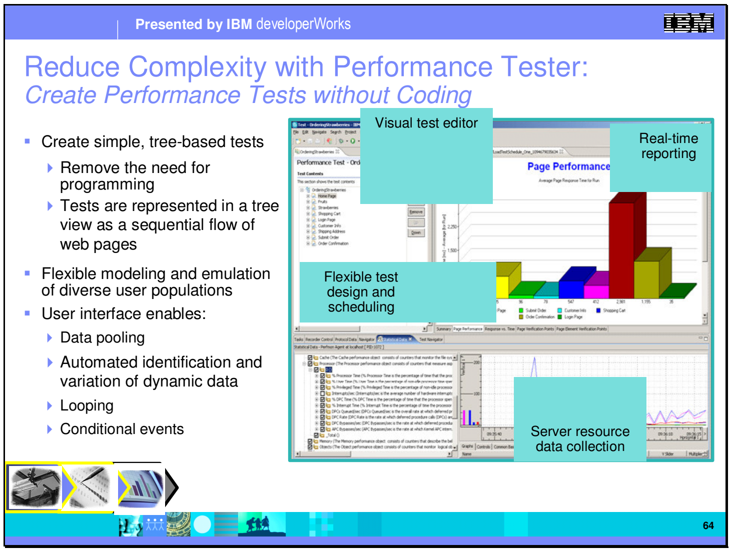
Rational Performance Tester addresses these issues in the following ways.

Tool Complexity. Creation of the scripts is made easy by the tool. You are primarily working within a graphical editor to both create the scripts and modify them. It will also automate the task of insuring data consistency by correlating the data used.

Flexibility. From a flexibility standpoint, the experienced tester can directly access the created script and modify it if desired. In the performance tester, Java Script is used.

Discovery and Diagnosis. The tool has powerful analysis and diagnostic capabilities which can help you understand the results and identify and pinpoint performance bottlenecks in J2EE applications.

Slide 64



Presented by IBM developerWorks

Reduce Complexity with Performance Tester: *Create Performance Tests without Coding*

- Create simple, tree-based tests
  - Remove the need for programming
  - Tests are represented in a tree view as a sequential flow of web pages
- Flexible modeling and emulation of diverse user populations
- User interface enables:
  - Data pooling
  - Automated identification and variation of dynamic data
  - Looping
  - Conditional events

Visual test editor
Real-time reporting
Page Performance
Flexible test design and scheduling
Server resource data collection
64

IBM Rational Performance Tester simplifies performance test creation, execution and analysis to help teams validate the ability of their applications to accommodate high multi-user loads before deployment. Introducing a variety of ease-of-use features to help teams rapidly detect system bottlenecks, Rational Performance Tester ensures both the novice and expert tester achieve day-one productivity, improving a team’s ability to validate Web scalability and reliability

Highlights:

Simple, tree-based tests – Underlying test code is hidden from the tester, ensuring day one productivity by even novice members of the test or IT operations team. However, custom Java code can be inserted by advanced testers to perform advanced data processing and analysis.

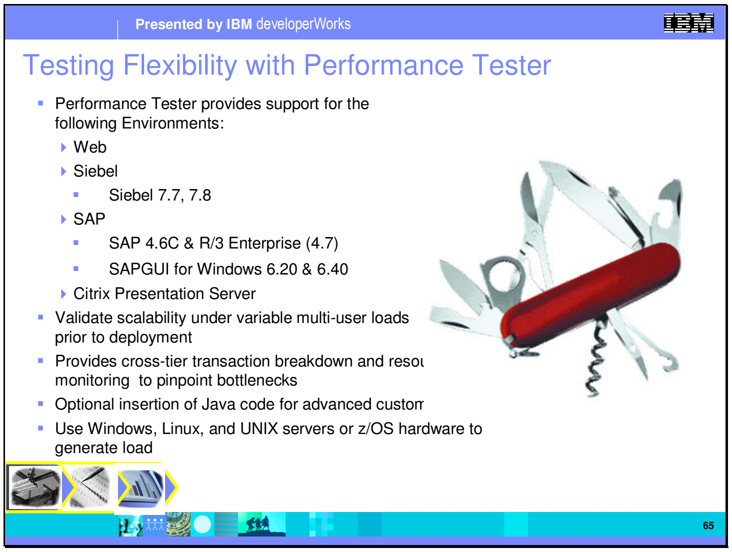
Flexible modeling – Crucial to successful performance testing is accurate replication of true user activity. The test scheduler contains a variety of customization options to increase the accuracy of the user simulations generated during test execution.

Automated identification and variation of dynamic data – All string values returned by the Web server or sent during test recording are detected and identified. During playback, strings returned by the Web server – typically different for each simulated user – can be analyzed independently using custom Java code. Strings entered by the user during test recording can be replaced by data pulled from datapools, enabling data-driven testing in which each simulated user “appears” unique to the server under test.

Real-time reporting – Web server response times are graphically displayed in real-time, ensuring live feedback that can be used for rapid detection of system bottlenecks. Additional server resource information (e.g. queue time, processor time) can be gathered and displayed – important for diagnosing hardware root causes.

Low memory and processor footprint – High emulated user loads can be achieved with existing hardware resources

Slide 65



Presented by IBM developerWorks

Testing Flexibility with Performance Tester

- Performance Tester provides support for the following Environments:
  - Web
  - Siebel
    - Siebel 7.7, 7.8
  - SAP
    - SAP 4.6C & R/3 Enterprise (4.7)
    - SAPGUI for Windows 6.20 & 6.40
  - Citrix Presentation Server
- Validate scalability under variable multi-user loads prior to deployment
- Provides cross-tier transaction breakdown and resou monitoring to pinpoint bottlenecks
- Optional insertion of Java code for advanced custom
- Use Windows, Linux, and UNIX servers or z/OS hardware to generate load

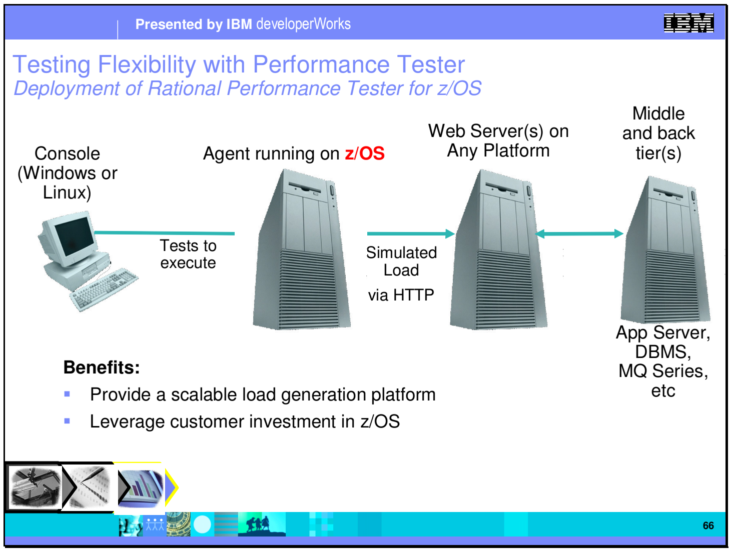
65

For Flexibility, we provide support for a range of test environments. Along with web based J2EE applications, we have support for Siebel, SAP, and Citrix applications.

Additionally, we can validate scalability under a range of multi-user loads. Also, we can give a breakdown of cross-tier transactions, monitoring the use of resources to pinpoint bottlenecks.

The use of standard Java Scripting along with support for a wide range of server environments – Windows, Unix, Linux or z/OS – further provides flexibility.

Slide 66



Presented by IBM developerWorks

Testing Flexibility with Performance Tester

Deployment of Rational Performance Tester for z/OS

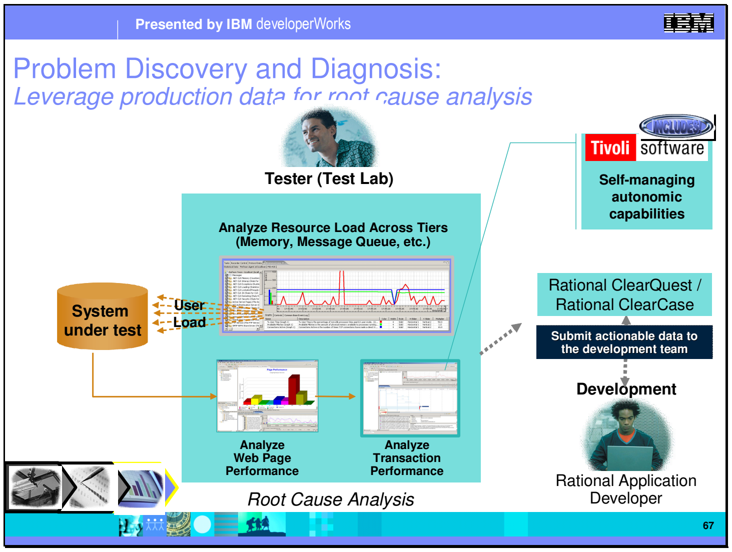
Console (Windows or Linux)
Tests to execute
Agent running on z/OS
Simulated Load
via HTTP
Web Server(s) on Any Platform
Middle and back tier(s)
App Server, DBMS, MQ Series, etc

Benefits:

- Provide a scalable load generation platform
- Leverage customer investment in z/OS

In this example, we are running our console client on either a Windows or Linux client, and have the test agent installed on a z/OS system. This allows us to generate very large loads using the z/OS machine, and protects your investment in z/OS by enabling its use as a test server.

Slide 67


Presented by IBM developerWorks

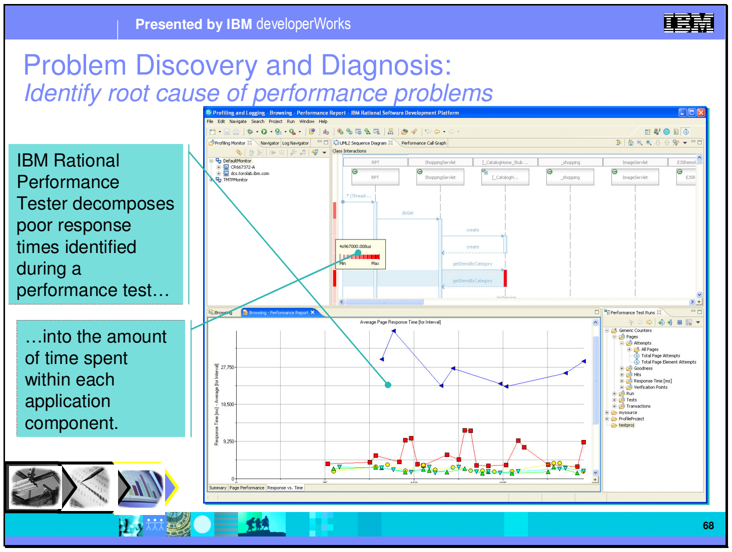
Problem Discovery and Diagnosis:
Leverage production data for root cause analysis

Tester (Test Lab)
Analyze Resource Load Across Tiers (Memory, Message Queue, etc.)
System under test
User Load
Analyze Web Page Performance
Analyze Transaction Performance
Root Cause Analysis
INCLUDES!
Tivoli software
Self-managing autonomic capabilities
Rational ClearQuest / Rational ClearCase
Submit actionable data to the development team
Development
Rational Application Developer
67

Rational Performance testers incorporate Tivoli's self-managing autonomic computing technology to provide the ability to perform detailed analysis and breakdown of J2EE transactions when a system is under load. This transaction breakdown helps testing teams to pinpoint the component responsible for the bottleneck.

For IT organizations that use Tivoli ITCAM solutions, this detailed performance data can be delivered directly to the developers desktop to help the developer pinpoint the source code causing the bottleneck.

Slide 68


Presented by IBM developerWorks

Problem Discovery and Diagnosis:

Identify root cause of performance problems

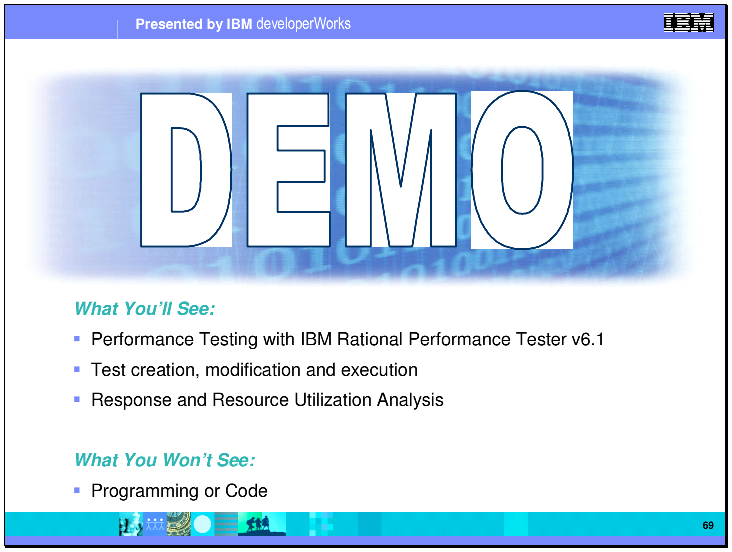
IBM Rational Performance Tester decomposes poor response times identified during a performance test…

…into the amount of time spent within each application component.

Profiling and Logging - Browsing - Performance Report - IBM Rational Software Development Platform
UML2 Sequence Diagram
Performance Call Graph
Average Page Response Time [for Interval]
Performance Test Runs
Summary
Page Performance
Response vs. Time

This free add-on toolkit for Rational Performance Tester allows for the rapid discovery of performance problems and the associated root causes, significantly reducing performance testing cycles.

Slide 69



DEMO

What You'll See:

- Performance Testing with IBM Rational Performance Tester v6.1
- Test creation, modification and execution
- Response and Resource Utilization Analysis

What You Won't See:

- Programming or Code

Slide 70

Presented by IBM developerWorks

Summarizing Key Features in Performance Tester

- **Built for Day 1 Productivity**
  - Create code free tests
  - Mask complexity to get the job done
- **Advanced Data Access & Manipulation**
  - Create code free schedules
  - Automated data variation and synchronization
- **Lower Cost of Performance Testing**
  - Low resource requirements
  - Desktop IDE integration

70

To wrap up, Rational Performance Tester can reduce complexity by giving productivity – Day 1. We do this by masking much of the complexity with an easy to use visual editor, letting you create code free tests. Flexibility is enhanced with automated data variation and synchronization, and creation of testing schedules with out coding.

The low resource requirements let you do large load simulations, and the desktop integration into the Eclipse IDE gives the needed integration with the development teams. All this helps to lower the cost of performance testing.

Slide 71

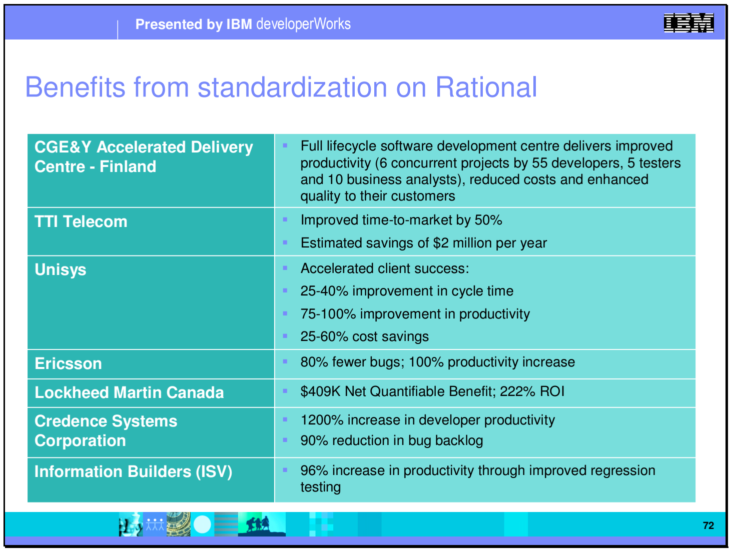
Presented by **IBM** developerWorks

Agenda

- Business Driven Development
- Test Management
- Code and Unit Testing
- Manual Functional Testing
- Automated Functional Testing
- Break
- Performance Testing
- **Wrap Up & Resources**

.

Slide 72



Presented by IBM developerWorks

Benefits from standardization on Rational

CGE&Y Accelerated Delivery Centre - Finland	▪ Full lifecycle software development centre delivers improved productivity (6 concurrent projects by 55 developers, 5 testers and 10 business analysts), reduced costs and enhanced quality to their customers
TTI Telecom	▪ Improved time-to-market by 50% ▪ Estimated savings of $2 million per year
Unisys	▪ Accelerated client success: ▪ 25-40% improvement in cycle time ▪ 75-100% improvement in productivity ▪ 25-60% cost savings
Ericsson	▪ 80% fewer bugs; 100% productivity increase
Lockheed Martin Canada	▪ $409K Net Quantifiable Benefit; 222% ROI
Credence Systems Corporation	▪ 1200% increase in developer productivity ▪ 90% reduction in bug backlog
Information Builders (ISV)	▪ 96% increase in productivity through improved regression testing

These IBM customers have bought into the IBM business driven development strategy for delivering quality software in support of their businesses; and standardized on the IBM software delivery tools, processes and best practices.

Cap Gemini Ernst & Young – Finland – This full lifecycle software development centre delivers improved productivity (6 concurrent projects by 55 developers, 5 testers and 10 business analysts), reduced costs and enhanced quality to their customers

TTI Telecom – Facing a business downturn in the telecom marketplace prompted TTI Telecom to look for innovative ways to reduce costs and pass on those savings to its customers. By standardizing on IBM Rational products, it successfully cut its time to market in half and realized cost savings estimated at $2 million per year.

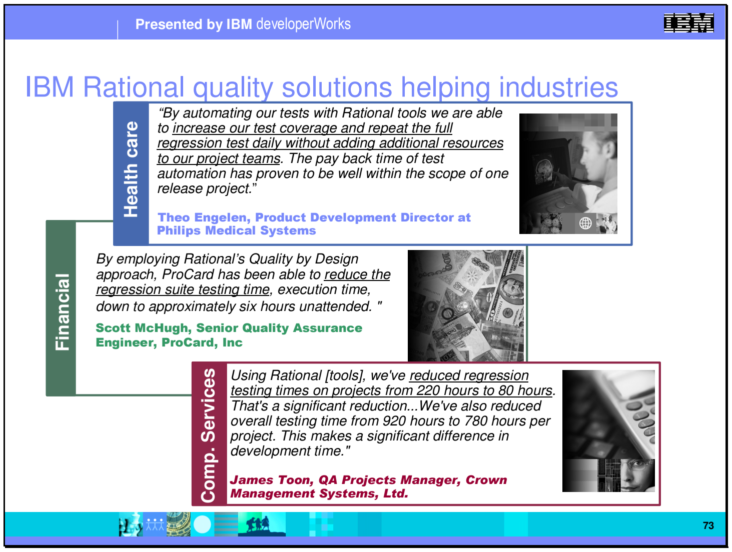
Ericsson has a longstanding commitment to the IBM Rational development platform. After standardizing on IBM Rational tools, this global telecommunications leader cut its standard production time from 16 to 12 months. It also realized a 30% reduction in required documentation, and reported 80% fewer bugs.

Lockheed Martin Canada realized a 222% ROI from standardizing on IBM Rational Suite Enterprise, IBM Rational ClearCase, IBM Rational ClearQuest, IBM Rational Rose, and IBM Rational Soda

Credence Systems Corporation, a manufacturer of automatic test equipment (ATE) (chip testers) first began using ClearCase and ClearCase MultiSite in 1997. In 2000 it standardized on IBM Rational Suite Enterprise. Credence’s lead architect “saw our development cycle go from roughly three years to about three months...That’s a tremendous return on investment.” 1200%.

Information Builders, Inc. is an Independent Software Vendor (ISV) of next-generation business intelligence systems. The company realized a 96% increase in productivity after implementing IBM Rational Suite TestStudio. Explains QA manager Steve Palmo, “Prior to using IBM Rational TestStudio and IBM Rational Robot it would take us months in order to get a project up to its quality standards. Now with regression testing capabilities in Robot we can actually test that same application in underneath a week.”

Slide 73



Presented by IBM developerWorks

IBM Rational quality solutions helping industries

Health care

"By automating our tests with Rational tools we are able to increase our test coverage and repeat the full regression test daily without adding additional resources to our project teams. The pay back time of test automation has proven to be well within the scope of one release project."

Theo Engelen, Product Development Director at Philips Medical Systems

Financial

By employing Rational's Quality by Design approach, ProCard has been able to reduce the regression suite testing time, execution time, down to approximately six hours unattended. "

Scott McHugh, Senior Quality Assurance Engineer, ProCard, Inc

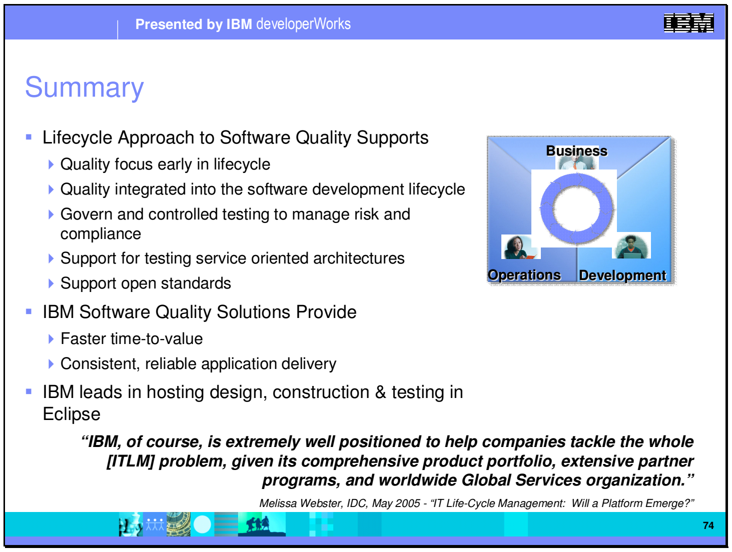
Comp. Services

Using Rational [tools], we've reduced regression testing times on projects from 220 hours to 80 hours. That's a significant reduction...We've also reduced overall testing time from 920 hours to 780 hours per project. This makes a significant difference in development time."

James Toon, QA Projects Manager, Crown Management Systems, Ltd.

These are some examples of how the IBM Rational quality solutions are helping companies in different industries.

Slide 74



Presented by IBM developerWorks

Summary

- Lifecycle Approach to Software Quality Supports
  - Quality focus early in lifecycle
  - Quality integrated into the software development lifecycle
  - Govern and controlled testing to manage risk and compliance
  - Support for testing service oriented architectures
  - Support open standards
- IBM Software Quality Solutions Provide
  - Faster time-to-value
  - Consistent, reliable application delivery
- IBM leads in hosting design, construction & testing in Eclipse

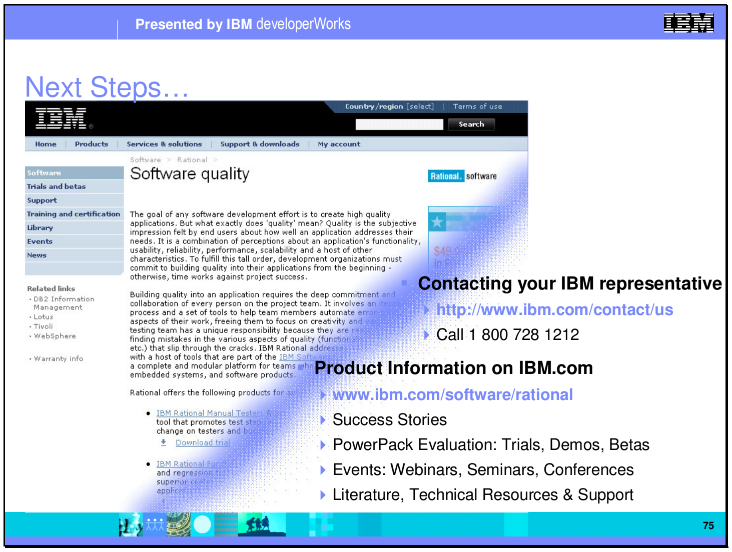
"IBM, of course, is extremely well positioned to help companies tackle the whole [ITLM] problem, given its comprehensive product portfolio, extensive partner programs, and worldwide Global Services organization."

Melissa Webster, IDC, May 2005 - "IT Life-Cycle Management: Will a Platform Emerge?"

In summary, we have looked at how a quality focus through the whole lifecycle of development gives greater software quality at a lower overall cost. The IBM Rational suite of quality products ranges from developer testing using unit testing and the profiling using the Eclipse IDE, defect and diagnostic testing with Purify Plus, and the three testing products Manual Tester, Functional Tester, and Performance Tester. We have also seen how governance plays a critical role in managing the process and reducing risk.

IBM leads in the hosting, design, construction, and testing in Eclipse and Windows platforms. This gives a faster time to achieving real value and consistent and reliable application delivery.

Slide 75


Presented by IBM developerWorks
IBM

Next Steps…

IBM
Country/region [select] | Terms of use
Search
Home | Products | Services & solutions | Support & downloads | My account
Software > Rational >
Software
Trials and betas
Support
Training and certification
Library
Events
News
Related links
· DB2 Information Management
· Lotus
· Tivoli
· WebSphere
· Warranty info
Rational. software
$49.0
in F

Software quality

The goal of any software development effort is to create high quality applications. But what exactly does 'quality' mean? Quality is the subjective impression felt by end users about how well an application addresses their needs. It is a combination of perceptions about an application's functionality, usability, reliability, performance, scalability and a host of other characteristics. To fulfill this tall order, development organizations must commit to building quality into their applications from the beginning - otherwise, time works against project success.

Building quality into an application requires the deep commitment and
collaboration of every person on the project team. It involves an itera
process and a set of tools to help team members automate error-
aspects of their work, freeing them to focus on creativity and va
testing team has a unique responsibility because they are res
finding mistakes in the various aspects of quality (functiona
etc.) that slip through the cracks. IBM Rational addresse
with a host of tools that are part of the IBM Softwa
a complete and modular platform for teams who
embedded systems, and software products.

Rational offers the following products for au

- IBM Rational Manual Tester: A
  tool that promotes test step
  change on testers and bu
  Download trial
- IBM Rational Fun
  and regression t
  superior
  applica

- **Contacting your IBM representative**
  - http://www.ibm.com/contact/us
  - Call 1 800 728 1212
- **Product Information on IBM.com**
  - www.ibm.com/software/rational
  - Success Stories
  - PowerPack Evaluation: Trials, Demos, Betas
  - Events: Webinars, Seminars, Conferences
  - Literature, Technical Resources & Support

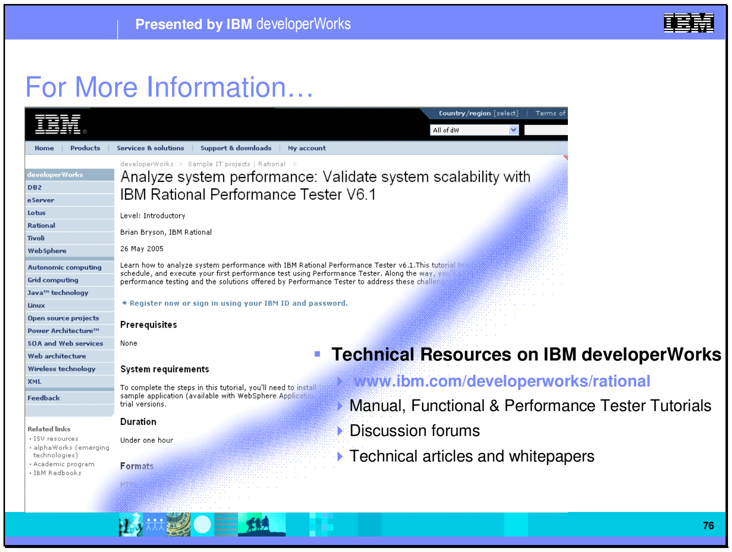
Success Stories

http://www-306.ibm.com/software/success/cssdb.nsf/topstoriesFM?OpenForm&Site=rational

PowerPack Evaluation Program

http://vrmnaweb.ibm.com/@^@DP0016110700000080707247/bin/devworks/entry.cgi?cl=USEN&pg=../../docs/devworks/lp1.html&id=48

Slide 76



For More Information…

developerWorks > Sample IT projects | Rational >

Analyze system performance: Validate system scalability with IBM Rational Performance Tester V6.1

Level: Introductory

Brian Bryson, IBM Rational

26 May 2005

Learn how to analyze system performance with IBM Rational Performance Tester v6.1.This tutorial tea schedule, and execute your first performance test using Performance Tester. Along the way, you'll e performance testing and the solutions offered by Performance Tester to address these challenge

Register now or sign in using your IBM ID and password.

Prerequisites

None

System requirements

To complete the steps in this tutorial, you'll need to install I sample application (available with WebSphere Applicatio trial versions.

Duration

Under one hour

Formats

HTML

- **Technical Resources on IBM developerWorks**
  - www.ibm.com/developerworks/rational
  - Manual, Functional & Performance Tester Tutorials
  - Discussion forums
  - Technical articles and whitepapers

For more information on testing and all aspects of software development, developerWorks is the place to go. It is IBM's main portal for developer information.