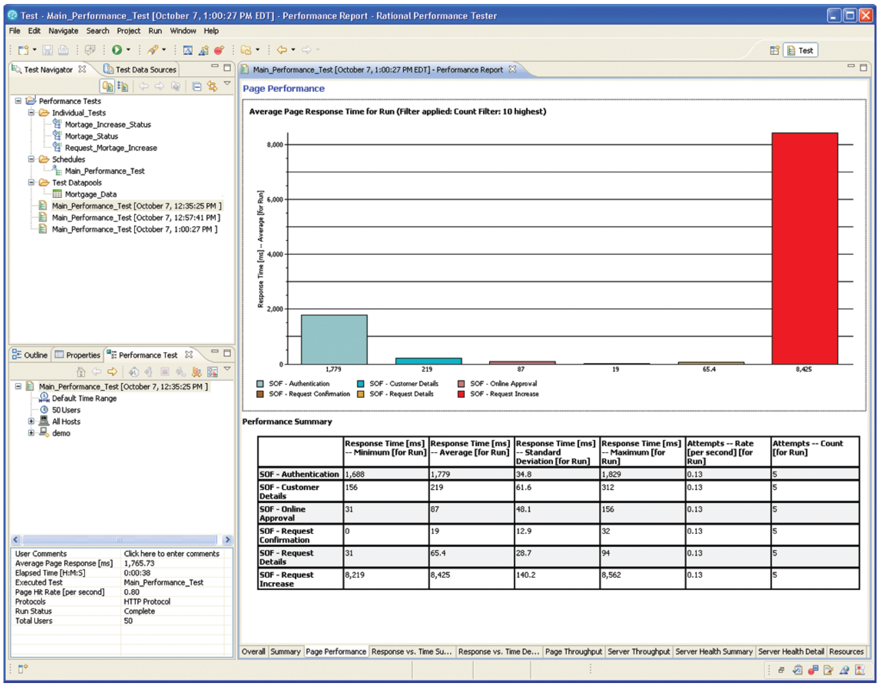
Rational. software

IBM Rational Performance Tester

Highlights

- ***Identifies the presence and cause of system performance bottlenecks***
- ***Enables you to create scriptless tests quickly, without writing or editing code***
- ***Provides a rich, tree-based test editor that delivers high-level and detailed views of tests***
- ***Helps ensure reliable playback with automated test data variation***
- ***Enables insertion of custom Java code for flexible test customization***
- ***Automates the identification and management of dynamic server responses***
- ***Collects and integrates server resource data with realtime application performance data***
- ***Minimizes the memory and processor footprint, enabling large multiuser tests with limited hardware resources***



User Comments	Click here to enter comments
Average Page Response [ms]	1,765.73
Elapsed Time [H:M:S]	0:00:38
Executed Test	Main_Performance_Test
Page Hit Rate [per second]	0.80
Protocols	HTTP Protocol
Run Status	Complete
Total Users	50

Performance Summary

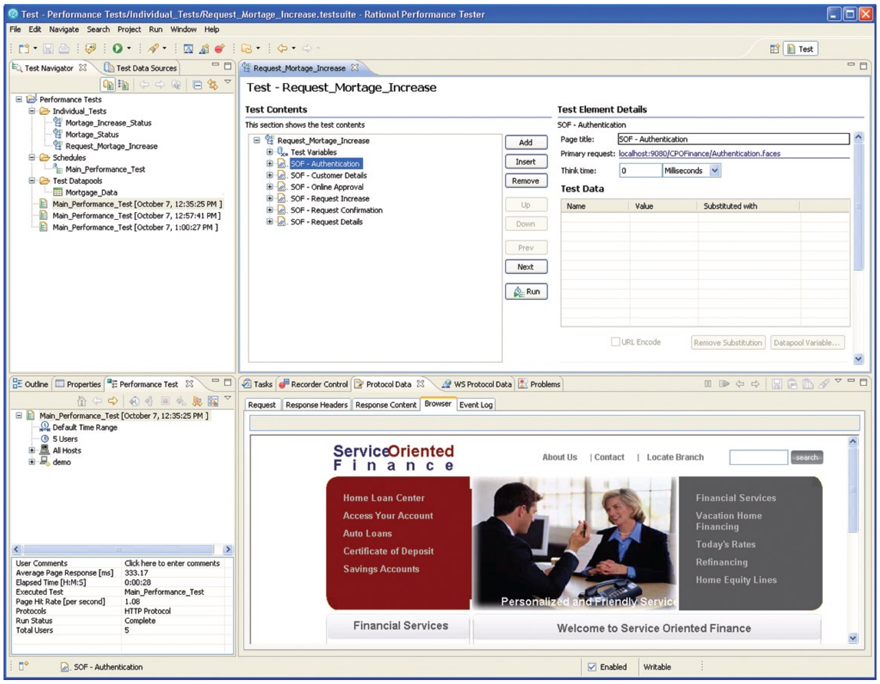
	Response Time [ms] -- Minimum [for Run]	Response Time [ms] -- Average [for Run]	Response Time [ms] -- Standard Deviation [for Run]	Response Time [ms] -- Maximum [for Run]	Attempts -- Rate [per second] [for Run]	Attempts -- Count [for Run]
SOF - Authentication	1,688	1,779	34.8	1,829	0.13	5
SOF - Customer Details	156	219	61.6	312	0.13	5
SOF - Online Approval	31	87	48.1	156	0.13	5
SOF - Request Confirmation	0	19	12.9	32	0.13	5
SOF - Request Details	31	65.4	28.7	94	0.13	5
SOF - Request Increase	8,219	8,425	140.2	8,562	0.13	5

Figure 1: Rational Performance Tester offers a series of reports that enable testers to quickly and easily identify the presence and cause of system performance bottlenecks.

A performance test creation, execution and analysis tool, IBM Rational® Performance Tester software enables teams to validate the scalability and reliability of their Web and enterprise resource planning (ERP) applications before deployment. The software offers a simple-to-use test recorder, automated data manipulation, advanced scheduling, realtime reporting and a highly scalable execution engine to help ensure that applications are prepared to handle large user loads.

Application failure due to performance-related problems is preventable with predeployment performance testing. After deploying applications, many teams experience trouble with availability, reliability and scalability. These difficulties typically occur as a result of one or more of the following deficiencies:

- *Load test inexperience, inhibiting the creation of performance tests*
- *Failure to anticipate and understand true user behavior*
- *Inability to quickly identify a bottleneck and determine the root cause of the problem*



Test - Performance Tests/Individual_Tests/Request_Mortage_Increase.testsuite - Rational Performance Tester

- Performance Tests
  - Individual_Tests
    - Mortage_Increase_Status
    - Mortage_Status
    - Request_Mortage_Increase
  - Schedules
    - Main_Performance_Test
  - Test Datapools
    - Mortgage_Data
  - Main_Performance_Test [October 7, 12:35:25 PM]
  - Main_Performance_Test [October 7, 12:57:41 PM]
  - Main_Performance_Test [October 7, 1:00:27 PM]

Test - Request_Mortage_Increase

Test Contents

This section shows the test contents

- Request_Mortage_Increase
  - Test Variables
  - SOF - Authentication
  - SOF - Customer Details
  - SOF - Online Approval
  - SOF - Request Increase
  - SOF - Request Confirmation
  - SOF - Request Details

Test Element Details

SOF - Authentication

Page title: SOF - Authentication

Primary request: localhost:9080/CPOFinance/Authentication.faces

Think time: 0 Milliseconds

Test Data

Name	Value	Substituted with

□ URL Encode

- Main_Performance_Test [October 7, 12:35:25 PM]
  - Default Time Range
  - 5 Users
  - All Hosts
  - demo

User Comments	Click here to enter comments
Average Page Response [ms]	333.17
Elapsed Time [H:M:S]	0:00:28
Executed Test	Main_Performance_Test
Page Hit Rate [per second]	1.08
Protocols	HTTP Protocol
Run Status	Complete
Total Users	5

SOF - Authentication

☑ Enabled Writable

Figure 2: Rational Performance Tester features a tree-based editor for scriptless test editing. This enables testers to quickly create tests without requiring any complex programming.

Constructed to address these concerns, Rational Performance Tester is designed to reduce load testing complexity while providing flexible workload scheduling and reporting capabilities, helping to ensure that teams of various skill levels can effectively identify the presence and cause of application performance bottlenecks prior to deployment.

Creating tests rapidly

Rational Performance Tester features capabilities explicitly designed to enable you to quickly build, execute and analyze the impact of load on your application environment.

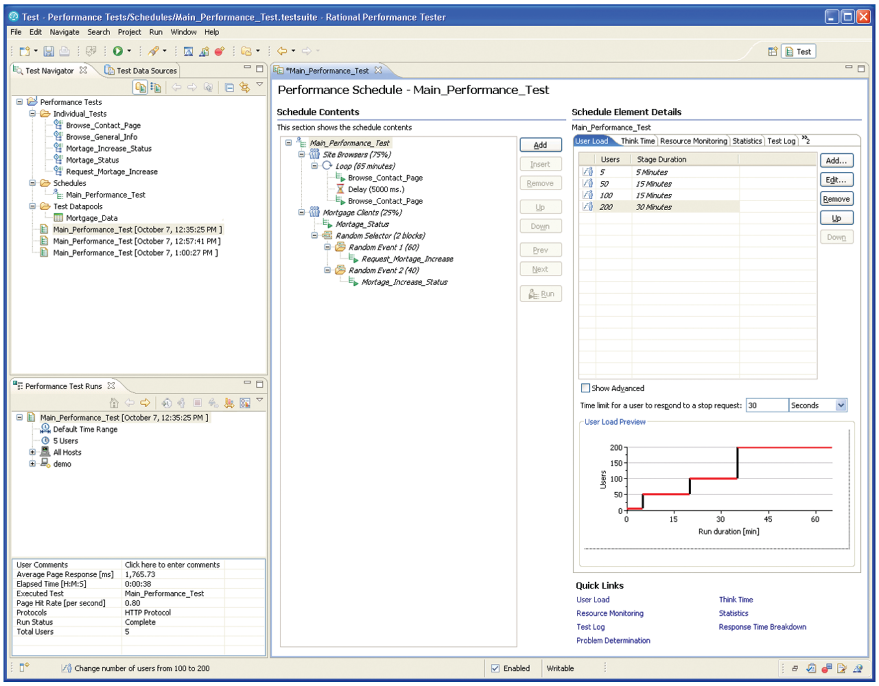
Test recording simply involves interaction with the targeted Web-based application or ERP solution through your browser or ERP client. The resulting test is presented in a concise, tree-based editor, masking code but exposing underlying details to the expert on an as-needed basis. Using the built-in application-specific viewer, you can view each visited page or screen offline to review user interaction and transaction ordering.

Understanding application data relationships

The interconnected nature of modern applications implies that data provided during one stage of a test is often passed along for use in subsequent stages. When conducting a performance test, you must maintain the integrity of these relationships in order for the test to be successful. Rational Performance Tester provides automated data relationship management capabilities that can identify and maintain application data relationships without requiring any mapping, test editing or other interaction from the user. This feature helps ensure reliable test execution and accurate load simulation.

Strategizing for accurate load emulation

To ensure a realistic load and prevent server caching, each virtual tester often needs to provide unique data during a test. As an example, to test the login functionality of a system, each virtual tester should access the system with a unique user ID and password. This method of data-driven testing is fully supported by Rational Performance Tester. The software automatically detects variable data entered during test recording and maps these items to rows and columns in spreadsheet-like data pools. During test execution, these pools are accessed, and unique data for each virtual tester is collected, helping to ensure accurate load emulation on the server.



Test - Performance Tests/Schedules/Main_Performance_Test.testsuite - Rational Performance Tester

```
Performance Tests
  Individual_Tests
    Browse_Contact_Page
    Browse_General_Info
    Mortage_Increase_Status
    Mortage_Status
    Request_Mortage_Increase
  Schedules
    Main_Performance_Test
  Test Datapools
    Mortgage_Data
  Main_Performance_Test [October 7, 12:35:25 PM ]
  Main_Performance_Test [October 7, 12:57:41 PM ]
  Main_Performance_Test [October 7, 1:00:27 PM ]
```

```
Main_Performance_Test [October 7, 12:35:25 PM ]
  Default Time Range
  5 Users
  All Hosts
  demo
```

User Comments	Click here to enter comments
Average Page Response [ms]	1,765.73
Elapsed Time [H:M:S]	0:00:38
Executed Test	Main_Performance_Test
Page Hit Rate [per second]	0.80
Protocols	HTTP Protocol
Run Status	Complete
Total Users	5

Performance Schedule - Main_Performance_Test

Schedule Contents

This section shows the schedule contents

```
Main_Performance_Test
  Site Browsers (75%)
    Loop (65 minutes)
      Browse_Contact_Page
      Delay (5000 ms.)
      Browse_Contact_Page
  Mortgage Clients (25%)
    Mortage_Status
    Random Selector (2 blocks)
      Random Event 1 (60)
        Request_Mortage_Increase
      Random Event 2 (40)
        Mortage_Increase_Status
```

Schedule Element Details

Main_Performance_Test

Users	Stage Duration
5	5 Minutes
50	15 Minutes
100	15 Minutes
200	30 Minutes

☐ Show Advanced

Time limit for a user to respond to a stop request: 30 Seconds

Change number of users from 100 to 200

Figure 3: Rational Performance Tester schedules enable you to graphically create realistic load schedules, which can optionally increase load at specific points to help you find your performance bottlenecks.

Validating server response

The IBM Rational Performance Tester interface provides you with a variety of options to validate server responses. With simple point-and-click actions, verification points can be inserted to validate server response codes, response sizes or portions of response content. For system-unique intricacies, Rational Performance Tester offers the ability to insert industry-standard Java™ code to address almost any situation.

Executing based on user group

Test scripts can be grouped in various combinations to create an accurate transaction volume for a given projected user population. Using the intuitive schedule-building interface, you can specify load in terms of user transaction volume or time. To help identify bottlenecks, you can set schedules to automatically increase load at specific intervals, and you can synchronize user events to simulate critical peak capacity activities.

Identifying the presence and cause of performance bottlenecks

During test execution, you can view renderings of live server responses, accompanied by simple-to-read, real-time reports that update throughout the test run.

System resource information such as CPU and memory utilization statistics can be collected from remote servers and correlated with response time and throughput data. Collected resource data is crucial for diagnosing which remote system router, Web server, application server or database server is responsible for detected delays, as well as for pinpointing the component (for example, CPU, RAM or disk) that is causing the bottleneck.

Once a bottleneck has been identified, Rational Performance Tester can map transaction response times by physical application tier, down to the application source code. This technology goes beyond bottleneck identification; it analyzes the root cause of the performance issue, and provides development teams with the information they need to resolve performance defects.

Lowering the cost of performance testing

Rational Performance Tester generates a low processor and memory footprint when emulating multiple users. As a result, high levels of scalability can be achieved even if the team does not have access to excessive computing power. In addition, test execution and system information retrieval can occur on IBM z/OS®, Microsoft® Windows® and Linux® software–based machines, optimizing a team's usage of existing hardware resources.

Ensuring system scalability before you deploy

Using IBM Rational Performance Tester, you can quickly and easily identify the presence and cause of system bottlenecks before system deployment. Designed for professional and novice testers to ensure quality-driven software delivery, Rational Performance Tester provides the system performance metrics required for you to make critical application release decisions.

For more information

To learn more about IBM Rational Performance Tester software, contact your IBM representative or IBM Business Partner, or visit:

ibm.com/rational

IBM Corporation
Software Group
Route 100
Somers, NY, 10589
U.S.A.

Produced in the United States of America
October 2008
All Rights Reserved

References in this publication to IBM products or services do not imply that IBM intends to make them available in all countries in which IBM operates.

The information contained in this documentation is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this documentation, it is provided "as is" without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this documentation or any other documentation. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of IBM software.