



IBM® Rational® Performance Tester
Version 9.0.1

Getting started now with Rational Performance Tester

Gunang Waney
March 2017



Getting started now with Rational Performance Tester 9.1.0

Contents

Chapter 1: Start with Rational Performance Tester.....	5
Start now.....	6
Chapter 2: Performance testing.....	11
Rational Performance Tester.....	12
Chapter 3: Installation details.....	13
Chapter 4: Testing workflow.....	15
Setting performance requirements.....	16
Verifying the recorded baseline at playback.....	16
Chapter 5: User load simulation.....	19
Putting more users into a test.....	20
Setting user load stages.....	20
Chapter 6: Test results.....	23
Glossary.....	25
Index.....	27

Chapter 1

Start with Rational Performance Tester

Topics:

- [Start now](#)

Start now with *performance testing* with Rational Performance Tester. Download and install the Rational Performance Tester *Starter Edition*.

Related tasks

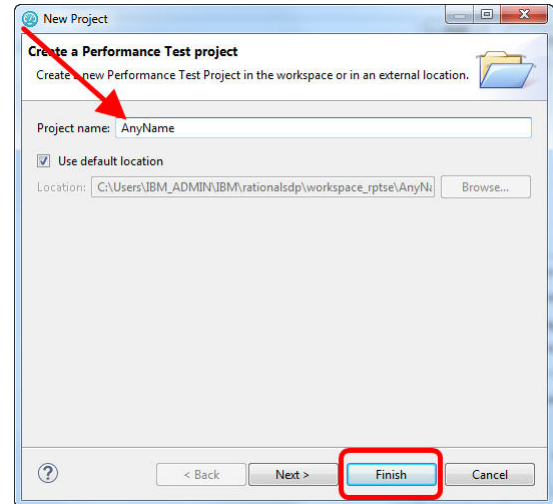
[Putting more users into a test](#) on page 20

[Verifying the recorded baseline at playback](#) on page 16

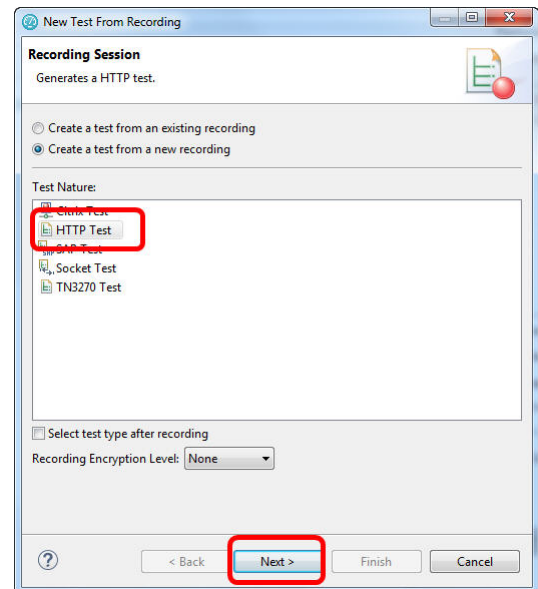
[Setting performance requirements](#) on page 16

Start now

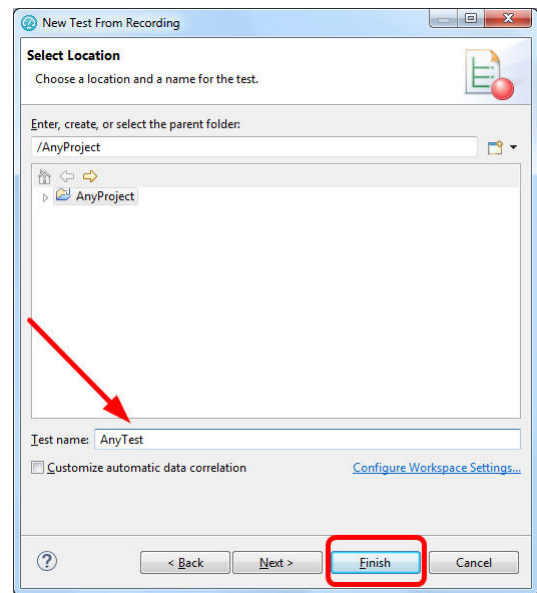
1. Download and install *Rational Performance Tester Starter Edition*.
2. Start Rational Performance Tester.
3. Close the **Welcome** screen.
4. Click **File > New > Performance Test Project**..
5. Enter a project name, any name, in the **New Project** window; then, click **Finish**.



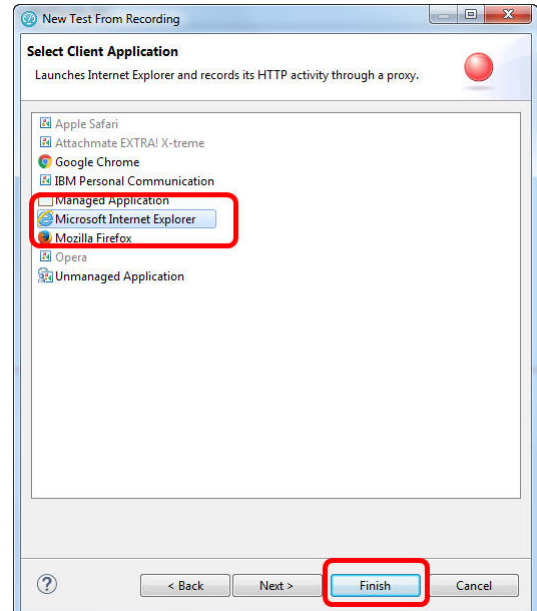
6. Click **HTTP Test** in **New Test From Recording**; then, click **Next**.



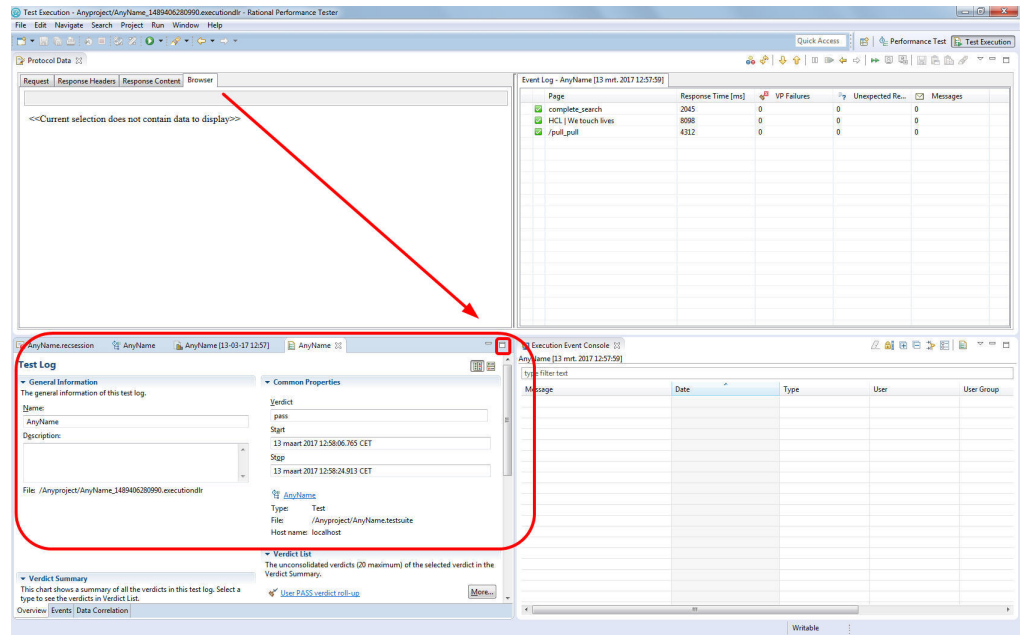
7. Enter a test name, any name. Then, click **Next**.



8. Click on Microsoft Internet Explorer or your default browser. Then, click **Finish**.



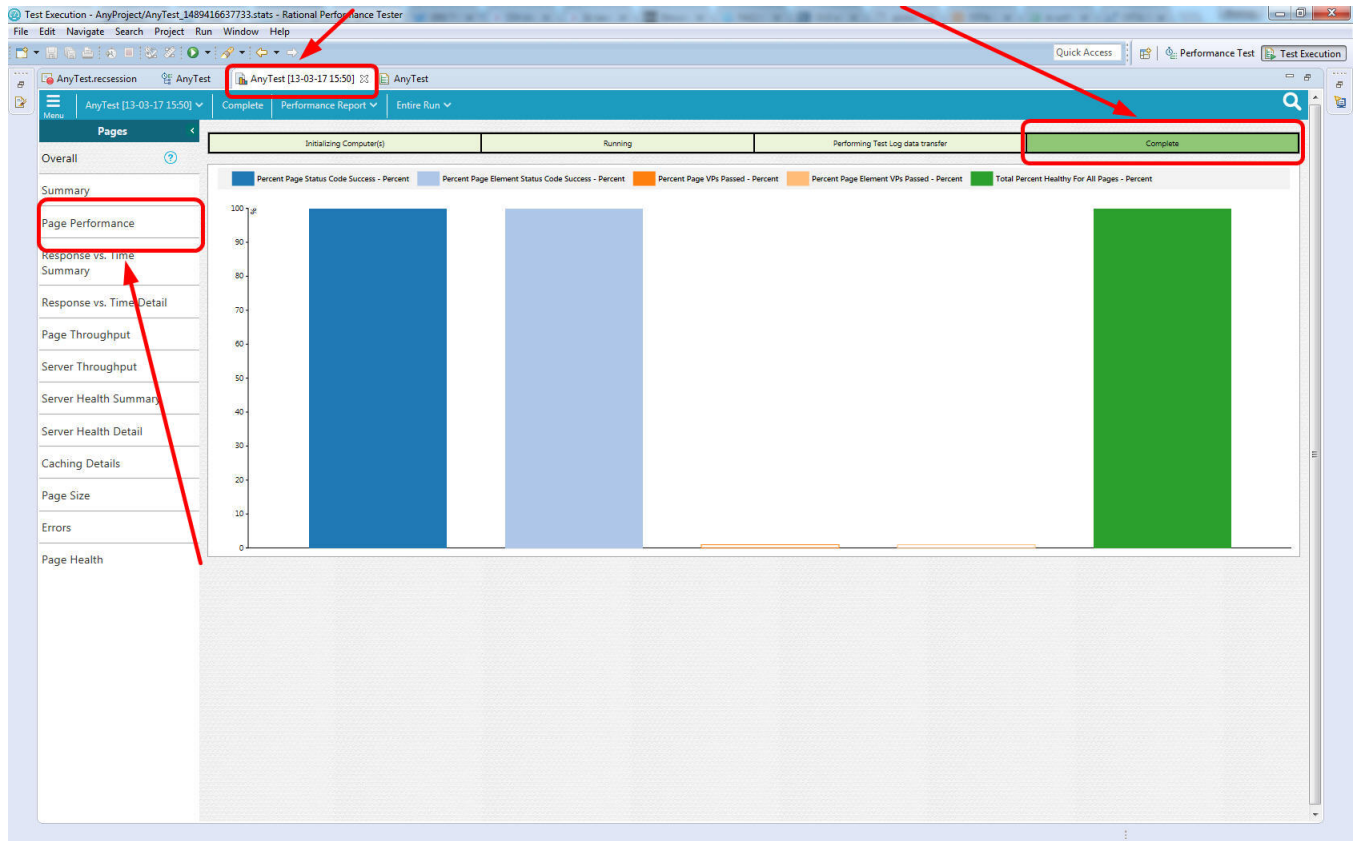
9. Wait until the browser opens with **Welcome to HTTP Test Recording**.
10. Enter your company's home page in the address field. "YourCompany.com"
11. Open any other page from the home page.
12. Close the browser (all tabs) after that page loaded fully.,
13. Click **Finish** in the **Domains Selection** window.
14. Wait until the test generation completes.
15. Click **Open test**; then, click **No** in the **Test Editor** dialog box.
16. Click **Run > Run**; then, click **Yes** to **Confirm Perspective Switch**.
17. Maximize the lower left quadrant.



18. Wait until the test completes. The indicator **Complete** must be green.

Ignore any correlation error message for now.

19. Click the tab with the timestamp.



20. Click **Page Performance** on the left side.

21. Review the graph of the **Average Page Response Time for Run**. See figure.

The response time of the first page is greater than the second page.

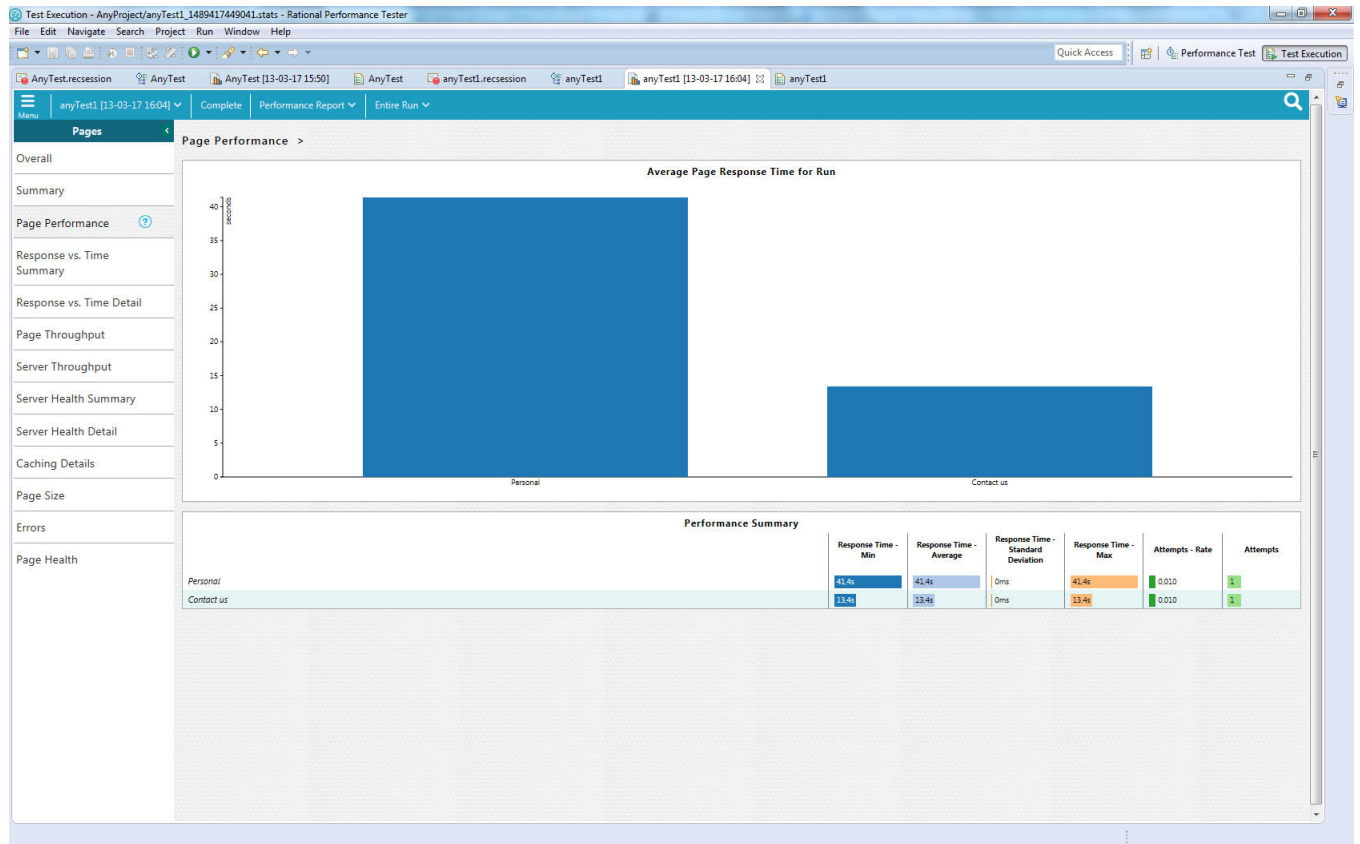


Figure 1: Average Page Response Time for Run

Related concepts

[Installation details](#) on page 13

[Rational Performance Tester](#) on page 12

[Performance testing](#) on page 11

[Testing workflow](#) on page 15

[Test results](#) on page 23

Chapter

2

Performance testing

Topics:

- [Rational Performance Tester](#)

Skipping *performance testing* is not an option. Before your web application goes live, you want to know whether your application meets the performance requirements.

In most cases, you want to know how your application behaves when a certain number of users connect to your application.

Afterward, when performance issues arise, you want to know the cause. Performance testing helps you to discover what factor determines the rate, the “bottleneck”.

Related tasks

[Start now](#) on page 6

[Setting performance requirements](#) on page 16

[Putting more users into a test](#) on page 20

Rational Performance Tester

Rational Performance Tester measures the network traffic between your client computer, and the application server. Through virtual users, you emulate the network traffic. You can run tests with multiple users through *performance schedules*.

To work with Rational Performance Tester, you don't need to know how to code.

Unlike functional tests, Rational Performance Tester tests do not depend on object recognition or the way you manipulate the GUI. Rational Performance Tester works with standardized network protocols. As such, the tool measures the time between the first byte out to the last byte in.

Rational Performance Tester can do the following network protocols.

- HTTP (browser-webserver traffic)
- Citrix
- SAP
- TN3270
- Socket
- SOA

Related tasks

[Start now](#) on page 6

[Setting performance requirements](#) on page 16

[Verifying the recorded baseline at playback](#) on page 16

Chapter

3

Installation details

The installation of Rational Performance Tester follows the same procedure as other IBM Rational products like Rational **Functional** Tester.

To get Rational Performance Tester installed quickly, you can download and install the [Rational Performance Tester Starter Edition](#).

Note: You must have administrative (Windows) or root (Linux) privileges to install Rational Performance Tester.

Watch the video [How to install IBM Rational Performance Tester Starter Edition](#) to install the Rational Performance Tester Starter Edition.

You can also read the [Installing Rational Performance Tester and Rational Performance Tester Agent](#) for a comprehensive description.

Related tasks

[Start now](#) on page 6

Chapter

4

Testing workflow

Topics:

- [Setting performance requirements](#)
- [Verifying the recorded baseline at playback](#)

You can follow the following workflow to work with Rational Performance Tester.

1. Determine the requirements.

Make clear what you want. How many users your application must handle. What the maximum response time of a request should be.

2. [Create a performance test project.](#)

3. [Record a test.](#)

Your computer issues requests to your application server,

4. Adjust the test to your requirements.

Edit the test in such a way that the test meets your needs. For example, set *verification points*.

5. Playback the test.

Your computer issues the same requests that you recorded and captures the responses from the server.

6. Assess the results.

Rational Performance Tester provides reports with graphs to support your analysis. For the individual events of the playback, open the *test log*.

Related tasks

[Start now](#) on page 6

Setting performance requirements

You can set limits to the accepted response times or the number of times that *verifications* must pass. To set the requirement so that the test fails when the test doesn't satisfy the requirement, select **Standard**.

1. Open the Performance Test perspective. Click **Window > Open Perspective > Other ... > Performance Test (default)**. Then click **OK**.
2. Follow the instructions in the following documents.
 - a) [Defining performance requirements in tests](#). See figure.
 - b) [Defining performance requirements in schedules](#)

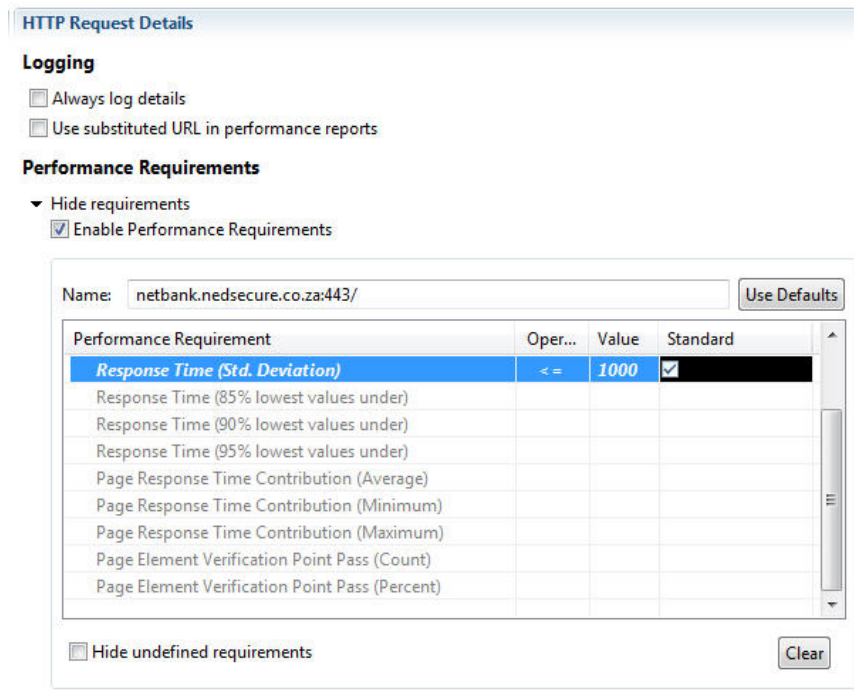


Figure 2: Performance requirements in tests

Related concepts

[Start with Rational Performance Tester](#) on page 5

[Rational Performance Tester](#) on page 12

[Performance testing](#) on page 11

Verifying the recorded baseline at playback

You can compare the following things in the playback actual with the recorded baseline through a *verification point*.

- Page title
- Response code
- Response size
- Response content

Image verification is available only for Citrix tests.

1. Open the Performance Test perspective. Click **Window > Open Perspective > Other ... > Performance Test (default)**. Then click **OK**.
2. Follow the instructions [Verifying expected behavior](#).

Related concepts

Start with Rational Performance Tester on page 5

Rational Performance Tester on page 12

Chapter 5

User load simulation

Topics:

- [Putting more users into a test](#)
- [Setting user load stages](#)

When users connect to your application server, the users impose a load on your server. Rational Performance Tester can simulate this situation through a *performance schedule*. You run a test with multiple virtual users.

In the real world, users don't connect to your web server at the same time. In a *performance schedule*, you can simulate the situation that the total number of users connect to your server in stages.

For more information, see [Emulating workloads](#).

Putting more users into a test

You can run tests with more users through a *performance schedule*. You can also add other options to run your test in a schedule.

1. Open the Performance Test perspective. Click **Window > Open Perspective > Other ... > Performance Test (default)**. Then click **OK**.
2. Create a performance schedule. Click **File > New > Performance Schedule..** Then, name the schedule and click **Finish**
3. Highlight the first item in the **Schedule Contents**.
4. Select **User Load** in the **Category** dropdown box of **Performance Schedule Details**.

You can assign up to 5 users to an HTTP schedule in the Rational Performance Tester Starter Edition. If you want to use more users, you need a separate virtual-users license.

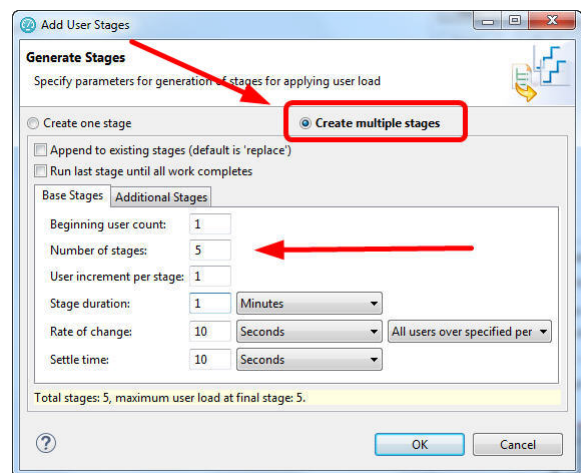
Related concepts

[Start with Rational Performance Tester](#) on page 5

[Performance testing](#) on page 11

Setting user load stages

1. Open the Performance Test perspective. Click **Window > Open Perspective > Other ... > Performance Test (default)**. Then click **OK**.
2. Create a performance schedule. Click **File > New > Performance Schedule..** Then, name the schedule and click **Finish**
3. Highlight the schedule.
4. Select **User Load** in the **Category** dropdown box, the default. Then, click **Add**.
5. Click **Create multiple stages** in the **Add User Stages** window.



6. Compose your stages. Then, click **OK**.

The **Performance Schedule Details** shows the number of users over the subsequent stages. See figure.

Performance Schedule Details

Category: **User Load**

Users	Stage Duration
1	10 Minutes
2	10 Minutes
3	10 Minutes
4	10 Minutes
5	<i>Until Finished</i>

Show Advanced

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

Percentage of users allowed to exit during execution: 0

Exit run for failing requirements

Number of failing stages in a row: 1

User Load Preview

General | **Advanced**

Figure 3: Increasing number of users in stages

Chapter

6

Test results

Rational Performance Tester creates execution reports and displays the reports automatically after a run.

Each test result begins with the name of the schedule or test, and ends with the timestamp of the run between square brackets.

The reports consist of figures and graphs that you can present to your manager or customer.

For more information, see [Viewing reports after a run](#).

If you want to see details about verification points and follow the course of the test, use the test log.

To open the test log, right-click the executed test in the **Test Navigator**; then click **Display Test Log**. For more information, see [Viewing test logs](#).

Related tasks

[Start now](#) on page 6

Glossary

Performance testing

Through performance testing, you measure the responsiveness and stability under a certain or various workloads. Performance testing comes in different types.

Load testing

Load testing is appropriate when you want to know what your application does under a specific load. For example, what are the response times when 100 users send requests to your application at the same time?

Stress testing

Stress testing indicates what load will cause your application to fail.

Soak or endurance testing

This test allows you to check what happens when your application suffers a certain load for a long time. A great way to see memory leaks.

Spike testing

In spike testing you increase the number of users suddenly with a significant amount. Then you see how the application reacts.

Configuration testing

What influences do different configurations have on your application?

Most users of Rational Performance Testing choose for load testing.

Verification point

A verification point is a point that you mark in the recording. The recorder saves the properties of your mark. You see a comparison of the recorded properties, or baseline, and the properties that Rational Performance Tester found during playback, or actual.

EXAMPLE

You record an HTTP test against your company website. You arrive at a login page with title “Login” and you want to make sure that the playback also arrives at this page. Then, you set a title verification point.

For details see [Specifying the expected page title](#).

Performance schedule

A performance schedule or schedule is a set that consists of one or more user groups that contains one or more tests. You can assign settings on all three levels. Highlight either the schedule, or user group, or test level. Then, discover the options on the right side of the screen.

The screenshot displays the 'Performance Schedule - Schedule' window. The interface is divided into several sections:

- Schedule Contents:** A tree view on the left showing a hierarchy: 'Schedule' > 'User Group 1 (100%)' > 'test'. A search box 'Enter filter text' is located above the tree.
- Options:** A vertical column of buttons: Add, Insert, Select, Remove, Up, Down, Prev, Next.
- Performance Schedule Details:** A panel on the right with a 'Category:' dropdown menu set to 'User Load'. Below it is a table with a 'User' column containing the value '5'. A list of categories is shown in a dropdown menu: Think Time, Resource Monitoring, Statistics, Variable Initialization, Performance Requirements, Test Log, Response Time Breakdown, and Problem Determination.
- Advanced Settings:**
 - Show Advanced
 - Time limit for a user to respond to a stop request: 30 Seconds
 - Percentage of users allowed to exit during execution: 0
 - Exit run for failing requirements
 - Number of failing stages in a row: 1
- User Load Preview:** A line graph showing 'Users' on the y-axis (0 to 4) and time on the x-axis (0 to 9). A red horizontal line is drawn at the level of 5 users.
- General/Advanced Tabs:** Located at the bottom of the details panel.
- Bottom Bar:** Includes 'Problems' (0 errors, 1 warning, 0 others), 'Protocol Data', and 'Recording Control'.

Figure 4: Schedule options in categories

Index

C

Citrix [12](#)

F

functional tests [12](#)

H

HTTP [6](#), [12](#), [20](#), [25](#)

L

license [20](#)

Linux [13](#)

N

network protocols [12](#)

P

page performance [8](#)

performance schedule [20](#), [20](#), [23](#), [25](#)

performance testing [25](#)

playback [15](#), [16](#), [25](#)

privileges [13](#)

project [6](#)

R

Rational Functional Tester [13](#)

recording [25](#)

response times [15](#), [16](#), [25](#)

S

SAP [12](#)

SOA [12](#)

Socket [12](#)

Starter Edition [13](#), [20](#)

T

test generation [7](#)

TN3270 [12](#)

V

verification point [23](#)

virtual user [12](#), [20](#)

W

Windows [13](#)

