

QALoad 05.05.01 Help

Getting Started, Running, and Analyzing Test Results

Technical support is available from our Customer Support Hotline or via our FrontLine Support Web site.

Customer Support Hotline:
1-800-538-7822

FrontLine Support Web Site:
<http://frontline.compuware.com>

This document and the product referenced in it are subject to the following legends:

Access is limited to authorized users. Use of this product is subject to the terms and conditions of the user's License Agreement with Compuware Corporation.

© 1998-2007 Compuware Corporation. All rights reserved. Unpublished - rights reserved under the Copyright Laws of the United States.

U.S. GOVERNMENT RIGHTS

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in Compuware Corporation license agreement and as provided in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii)(OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14 (ALT III), as applicable. Compuware Corporation.

This product contains confidential information and trade secrets of Compuware Corporation. Use, disclosure, or reproduction is prohibited without the prior express written permission of Compuware Corporation.

Compuware, ActiveAnalysis, ActiveData, Interval, QACenter, QADirector, QALoad, QARun, Reconcile, TestPartner, TrackRecord, and WebCheck are trademarks or registered trademarks of Compuware Corporation.

Acrobat® Reader copyright © 1987-2007 Adobe Systems Incorporated. All rights reserved. Adobe, Acrobat, and Acrobat Reader are trademarks of Adobe Systems Incorporated.

ICU License - ICU 1.8.1 and later COPYRIGHT AND PERMISSION NOTICE Copyright (c) 1995-2003 International Business Machines Corporation and others All rights reserved. Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

All other company or product names are trademarks of their respective owners.

US Patent Nos.: Not Applicable.

Doc. CWQLHX551
March 2, 2007

Table Of Contents

Welcome to QALoad	1
Getting Started	3
The Load Testing Process.....	3
QALoad Conductor.....	3
QALoad Player	3
System Under Test	3
Developing Scripts.....	4
Using QARun Scripts for Load Testing.....	4
Validating Scripts.....	4
Starting the Conductor.....	4
Setting up the Conductor	5
Running a Load Test	6
Preparing for a Load Test	6
Running a Load Test	6
Running a Series of Tests.....	6
Monitoring a Load Test	7
Stopping a Load Test.....	7
Adding Post-test Comments.....	7
Adding Post-test Comments.....	7
Analyzing Test Results	9
Analyzing Test Results.....	9
Custom Reports	9
Pre-defined Reports.....	9
Graphing Data	9
Managing Large Amounts of Data.....	10
QALoad Analyze Graph Types.....	10
Customizing Graphs.....	10
Integrating ServerVantage Agent Data	11
Displaying ServerVantage Agent Data.....	11
Index	13

Welcome to QALoad

With QALoad, you can simulate the load generated by thousands of users without the expense of actual end users or their equipment. QALoad enables you to quickly develop test scripts, control the conditions for the test, create the virtual users that simulate the load, initiate and monitor the test, and report the results.

The following components of QALoad control different aspects of the load testing process:

[Script Development Workbench](#)

Develop your scripts. Capture sessions and convert the transactions into C, C++, or Java-based script.

[Conductor](#)

Control session activity. Initiate and monitor the test.

[Player](#)

Simulate user activity based on commands given from the Conductor.

[Analyze](#)

View summary report data and create other statistical reports from the test.

Before you begin using QALoad, review the [Getting Started](#) section of the online help to familiarize yourself with load testing concepts.

Getting Started

The Load Testing Process

You begin the testing process by determining the types of application transactions you want to emulate. You then develop these transactions into QALoad scripts by creating the same types of requests that your applications invoke on the server. Each transaction becomes its own script. The QALoad Script Development Workbench lets you easily create full-function scripts.

When you plan your test, you need to decide which transactions to run, the number of simulated clients that will run each transaction script, and the frequency at which each script will run. When you run the test on a workstation with the QALoad Player component, you can specify transaction rates as well as fixed and random delays to better emulate real-system activity. QALoad considers these factors a test scenario and stores them in a session ID file.

While a test is running, the test operator can dynamically view overall run times as well as individual transaction performance. QALoad's Conductor component collects this data for analysis at the conclusion of the test.

After executing the test, summary reports show the response times that the emulated clients experienced during the test. Individual and global checkpoints let you view and identify specific areas of system performance. You can export all test output data to spreadsheet and statistical packages for further analysis or use QALoad's Analyze component to create presentation-quality reports and graphs.

As shown in the following image, a typical load test setup consists of a QALoad Conductor, one or more QALoad Players, and the system under test.

QALoad Conductor

QALoad's Conductor controls all testing activity such as setting up the session description files, initiating and monitoring the test, and reporting and analyzing test results.

QALoad Player

A QALoad Player creates virtual users that simulate multiple clients sending middleware calls to a server under test. In a typical test setup, one or more QALoad Player workstations run under any Windows 32-bit platform (Windows XP or 2000) or UNIX. For large tests (thousands of simulated clients), you can connect multiple Players to QALoad's Conductor. The Conductor and Players communicate using TCP/IP.

The hardware and software capabilities of the Player machine are the only factors that limit the capacity of an individual QALoad Player. The maximum number of virtual users per Player machine is dependent on the system under test, the characteristics of the script, and the test scenario. You can specify how many threaded- and process-based virtual users to assign on a machine on the [Machines tab](#) of the Options dialog box in the Conductor. The Conductor calculates how many virtual users will be active per 64 MB of RAM, based on the values you provide in these fields.

System Under Test

The servers you test are typically production systems or a duplicate of a production system that is set up at a test facility. If you perform any kind of system selection or performance stress test, the system under test

must use the same hardware and software (including current versions) as the production environment. Compuware has found that even subtle changes have profound effects on performance.

Developing Scripts

Use the QALoad Script Development Workbench to develop test scripts. It contains facilities for capturing sessions, converting captured sessions into scripts, and modifying and compiling scripts. Once you compile your script, use QALoad's Conductor and Player components to test your system.

Record Facility: QALoad's Record facility, which you can access through the QALoad Script Development Workbench, records the transactions that your terminal, browser, or client makes. It stores these transactions in a capture file.

Convert Facility: QALoad's Convert facility, which you can access through the QALoad Script Development Workbench, converts capture files into scripts. It generates a one-to-one correspondence of transactions from the original session to your QALoad script.

Visual Navigator: Visual Navigator for WWW is QALoad's easy-to-use visual interface to QALoad's powerful script development tools. Visual Navigator for WWW renders your recorded C-based transaction in a tri-paned, browser-like environment similar to popular visually-oriented development tools, with icons representing all the elements of your script.

Using QARun Scripts for Load Testing

QALoad provides you with the functionality to perform load tests using your QARun scripts. By inserting your QARun script into a QALoad script template, you can time your GUI-driven business transactions and include those timings in QALoad post-test reports.

Validating Scripts

Before you conduct an actual load test, you should individually validate the script(s) you plan to use by running a simple load test. If the script runs to the end without any errors and runs multiple times without errors, it is valid to use in a load test.

If the script aborts on an error, debug the script and run it through a simple test again. You can validate scripts from the QALoad Script Development Workbench, the QALoad Player, or the QALoad Conductor:

[Validating a Script in the Script Development Workbench](#)

[Validating a Script in the Player](#)

[Validating a Script in the Conductor](#)

Starting the Conductor

The following procedure describes how to start the Conductor.

To start the QALoad Conductor:

1. **From Windows:**

- Click Start>Programs>Compuware>QALoad>Conductor.

From a Command Prompt:

- Type `mpwin32 <session_file_name> /l /e /a /t`

The applicable parameters are defined in the following table.

Parameter	Definition
/l (Optional)	Creates a log file showing error messages and test status.
/e (Optional)	Exits the Conductor when the test completes.
/a (Optional)	Launches Analyze when the test completes.
/t (Optional)	Executes Conductor at a set time. Valid time formats are /txx:xx or /txx/xx/xx /txx:xx.

2. When the **Session Options** dialog box appears, select the appropriate option to create a new session ID file or to open an existing one. Click **OK**.

3. If you chose to open an existing session ID file, the Open dialog box appears. Navigate to the session ID file you wish to open, select it, and click **Open**. The selected session ID file opens in the Conductor Test Information Window.

If you chose to set up a new session ID file, the Conductor opens displaying the Test Information Window, and the configuration and setup toolbar. For information about the Conductor's configuration and setup toolbar buttons, see [Configuration and Setup Toolbar Buttons](#).

Setting up the Conductor

To prepare for running a load test, you must set up the Conductor.

To set up the Conductor:

1. [Start the Conductor](#).
2. [Configure the Conductor](#). After starting the Conductor, you may need to verify that the Conductor's configuration parameters are set properly.
3. [Set up a session ID file](#). For every test you run, you must create a session ID file containing information the Conductor needs to run the test, such as which scripts to run, which Player machines to use, and whether to collect server or performance monitoring data. You use the Conductor to create and save session ID files in the `\QALoad\Session` directory.

Running a Load Test

Preparing for a Load Test

Before you run a load test, you must complete the following tasks:

- ! **Prepare a datapool file:** If you created a datapool file using the QALoad Script Development Workbench, QALoad stores the file where the Conductor can automatically access it. However, if you created a datapool file using a text editor (for example, Notepad), you must place the file in your appropriate \Middleware\<middleware_name>\scripts directory (for example, \QALoad\Middleware\Oracle\Scripts) so the Conductor can access the file.

For information about datapool files, see [Simulating User-entered Data](#).

- ! **Set Up SSL Client Authentication for Virtual Users (SSL scripts only):** If you are running a load test with a WWW script containing SSL requests, you should export a Client Certificate from your browser into QALoad or create a QALoad Client Certificate for each virtual user that runs the script. This setup facilitates a one-to-one ratio of Client Certificates to virtual users, which more realistically simulates your testing environment.

To export Client Certificates from your browser and convert them for use in QALoad or to create QALoad Client Certificates, see [Importing a Client Certificate from a Web Browser](#).

Once you export or create the necessary Client Certificates, you can insert them into your script using datapools.

Running a Load Test

After validating a script using one of the methods described in [Validating Scripts](#), it is safe to run a load test with that script. See the following topics for more information:

- ! [Preparing for a Load Test](#)
- ! [Monitoring a Load Test](#)
- ! [Stopping a Load Test](#)

To start a load test:

Click the Run button  on the configuration and setup toolbar, or from the Conductor's Run menu, choose Start Test. While a test is running, the [Conductor's Interface](#) changes to provide you with real-time test options.

 **Note:** While any window on the desktop is re-sizing or re-positioning, all Windows applications pause. Do not click and hold on a window caption or border for extended periods during a load test because it delays message handling and may have an impact on test results.

While a load test is running, the Conductor's toolbar changes from the Configuration and Setup Toolbar to the Runtime Toolbar. The Runtime Toolbar buttons let you control the test and access detailed information about the test while it is running. For more information about what to expect from the QALoad Conductor while a test is running — including descriptions of the Runtime Toolbar buttons — see [Monitoring a Load Test](#).

Running a Series of Tests

You can also run a series of tests — a batch test. A batch test comprises multiple session ID files. When you run a batch test, the session files are executed sequentially until all of them are executed. The Conductor enables you to run multiple batch tests without operator intervention.

Monitoring a Load Test

When you start a test, the QALoad Conductor's interface changes to an interactive test control station, referred to as the **Runtime Window**. The Runtime Window displays information about the scripts, machines, and virtual users that are executing the load test. From the Runtime Window, you can observe the progress of individual scripts and Player machines, create and view real-time graphs, and start or suspend scripts and Players from a running test to better simulate the unpredictability of real users.

In addition to the test data shown by default on the Runtime Window, you can access detailed test information using the QALoad Conductor's **Runtime Toolbar Buttons**. You can:

- ! View statistics for a single virtual user
- ! View the activities of a virtual user in a browser-like window (WWW only)
- ! Step to the next request (WWW only)
- ! View the current datapool record
- ! Display the script running on a single virtual user
- ! Display messages sent from a Player workstation to the QALoad Conductor
- ! Display statistics about Conductor/Player communication
- ! Show/hide the Runtime Tree or Runtime Control Panel
- ! Synchronize all virtual users
- ! Exit, abort, or quit the test

Stopping a Load Test

A load test is complete when all virtual users exit. A virtual user automatically exits when one of the following occurs:

- ! A script encounters an EXIT command.
- ! A script completes its transaction loop.

To stop a load test, click the Exit button.

Adding Post-test Comments

If you selected the Display Post Test Comments option on the General tab of the Options dialog box when you configured the Conductor, the Post Test Comments window opens when you click the Quit button. Type any comments, which are saved to the test's Summary Report, which can be viewed in QALoad Analyze.

Adding Post-test Comments

By setting the appropriate options when you configure the Conductor, you can add comments after running a load test.

If you selected the Display Post Test Comments option on the General tab of the Options dialog box when you configured the Conductor, the Post Test Comments window opens when you click the Quit button. Type any comments, which are saved to the test's Summary Report, which can be viewed in QALoad Analyze.

Analyzing Test Results

Analyzing Test Results

After you set up a load test and run it, you can analyze the results from the test using QALoad Analyze.

An important part of the load testing process is viewing and studying the results of a test. You can view load test results not only on a machine where QALoad is installed, but on any machine with a Web browser.

When running a test using a particular session ID file (set up in the Conductor), each Player compiles a local timing file that comprises a series of timing records for each checkpoint of each script run on that Player. Each timing record in the file consists of a response time/elapsed time pair of values. These values specify the amount of time it took a certain checkpoint to finish (response time) at a specific time in the test (elapsed time).

At the end of a test, Player timing files are sent to the Conductor and are merged into a single timing file, called the Primary timing file, for analysis. If you set up integration with Compuware's ServerVantage product, the Conductor collects timing data from the ServerVantage central console and also merges that data into the timing file.

When you open a timing file, QALoad generates a working folder that contains all supporting files, reports, and images generated from that timing file. The folder is located in the `\Program Files\Compuware\QALoad\TimingFiles\xxx.xml.source` directory, where `<xxx>` is the name of the timing file.

Custom Reports

Use QALoad Analyze to create custom reports using XML (Extensible Markup Language), XSL (Extensible Style Language), and HTM (Hypertext Markup) files. QALoad Analyze provides a set of files in .htm, .xml, and .xsl formats in addition to the .tim file. QALoad Analyze automatically generates a XML (*.xml), XSL (*.xsl), and HTM (*.htm) file when you open a timing file.

Pre-defined Reports

QALoad Analyze provides pre-defined reports so you can receive immediate load test results without manipulating any data. All the files necessary for these reports are located in the directory `\Program Files\Compuware\QALoad\Timing Files\Reports`.

 Note: The pre-defined reports that are available depend on the data collected in the timing file. This is determined by the QALoad Conductor option you select when you run the load test.

Graphing Data

Starting with the Workspace, you can use QALoad Analyze's charting features to graph timing data in a number of formats and styles.

Managing Large Amounts of Data

With a large number of virtual users, it's possible to create a timing file containing hundreds of thousands of timing records for each checkpoint. Attempting to graph just a few of those checkpoints can slow QALoad Analyze down considerably. For example, if a timing file contained 250,000 timing records for each data point, attempting to graph even one checkpoint means that QALoad Analyze must paint 250,000 lines on the graph. Since most monitors only have 1024 pixels across the screen, the 250,000 data points would mostly be plotted atop one another and the results would be unreadable.

Now imagine attempting to graph the data of several data points of that size. The sheer amount of data could easily overwhelm a workstation. QALoad Analyze must also re-draw the graph every time you move the window, resize the window, right-click on the graph, and so on. You could conceivably spend enormous amounts of time simply attempting to graph data.

To make large amounts of data manageable, QALoad Analyze provides an option that enables you to determine how to thin data. That is, how to determine how many data points to plot.

QALoad Analyze Graph Types

The following basic graph types are available from QALoad Analyze. After generating one of the following graph types, you may further customize it in a number of ways.

Line Graph

A line graph plots response times versus elapsed times for the selected checkpoints. It provides a good representation of how much fluctuation there is in response times over the course of a test.

Bar Graph

A bar graph shows the median, mean, or percentile response times for the selected checkpoints.

Transaction Throughput Graph

This type of graph shows the cumulative number of transactions that occurred within the user-specified time range over the duration of the test.

Response Time Distribution Graph

This type of graph shows the percentage of checkpoint timings that fall within a particular response time range. A response time distribution graph shows if response times tend to fall within the range or are widely dispersed. A response time distribution graph only shows results for a single checkpoint, although it can compare results from multiple timing files.

Cumulative Response Time Distribution Graph

This type of graph shows the percentage of transactions for a single checkpoint that have a response time equal to or less than a specified value.

Customizing Graphs

You can change the style and appearance of a graph using options available from the Graph toolbar. Display the Graph toolbar by right-clicking in an open area of a graph and choosing the Toolbar option

from the shortcut menu. The Graph toolbar contains buttons for standard Windows operations as well as for customizing the appearance of your graphs.

The Graph toolbar includes the following features for customization:

- ! Graph type (gallery type)
- ! Color
- ! Grid orientation (horizontal and vertical)
- ! Legend box
- ! Data display
- ! Dimension (3D or 2D)
- ! Rotation
- ! Z-Cluster
- ! Color/pattern

Integrating ServerVantage Agent Data

If you set options to integrate ServerVantage resource utilization data before running a test, that data is included in the resulting timing file. It can be sorted and displayed in QALoad Analyze in much the same way as QALoad timing data. ServerVantage data provides a summary of all the Agents that ServerVantage monitored during the load test and details aggregate statistics for Agent data points including minimum, maximum, and mean data values.

Displaying ServerVantage Agent Data

When you open a timing file containing ServerVantage Agent data, QALoad Analyze displays test data with QALoad timing data two ways:

- ! ServerVantage Agent workstations are listed in the Server Monitoring group in the Workspace tree-view, under the Resource Trends (ServerVantage) branch. From the Workspace, select **Agent workstations** to create detail or graphical views of the Agent data points. Specifically, you can:
 - Display Agent data point details.
 - Graph Agent data point details.
- ! Detailed data point information is displayed in the Data window. The ServerVantage detail view includes data such as the name of the machine where you ran the ServerVantage Agent; the Agent name; and the minimum, maximum, and mean data values for the Agent.

 Note: ServerVantage resource utilization data is available only if you set the ServerVantage integration options on the QALoad Conductor's Test Information window before executing a load test.

Index

C

client certificate
in a datapool 6

Conductor

setting up 5
starting..... 4

D

datapool
client certificate 6

G

graph
types..... 9

I

Integrating ServerVantage Agent Data 11

L

load test

monitoring 7
preparing 6
running..... 6
stopping..... 7

load test 3

S

script

developing 4
validating..... 4

T

technical support iii

test

results

analyzing 9
running..... 6