QALoad 05.06

Getting Started, Running a Load Test, and Analyzing Results

Customer Support Hotline:
1-800-538-7822

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

Doc. CWQLHX560
November 8, 2007

Table Of Contents

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++ 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++ 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

Using Microsoft Windows Vista

If you are using QALoad on Microsoft Windows Vista, you must make the following configurations for proper operations of capturing in Winsock, performance monitoring, and Remote Monitoring for SNMP, Windows Registry, and WMI.

> Note: On the Microsoft Windows Vista platform, QAload currently only supports WWW and Winsock middlewares.

Winsock Capture

Winsock capture under Vista with Internet Explorer 7.0's Protected Mode enabled is not supported. Attempting to capture with Protected Mode enabled results in an error. Follow the steps below for instructions on how to disable the Protected Mode.

1. Open a browser window and select **Tools>Options** to open the Internet Options dialog box.

2. Select the **Security** tab. Clear the check box next to **Enable Protected Mode (requires restarting Internet Explorer)**.

3. Click **Apply** to save the change and **OK** to close the dialog box.

Performance Monitoring Group

In Vista, access to performance monitoring attributes is prohibited unless you are an administrative user or part of the Performance Monitoring Group. Any standard user attempting to use QALoad must be part of this group.

SNMP Remote Monitoring

In Vista, to enable SNMP remote monitoring, you must perform the following steps:

1. From the Windows task bar, click **Start>Settings>Control Panel>Programs and Features**.

2. In the Tasks list, select **Turn Windows features on or off**. Select the **SNMP feature** check box on the Windows Features dialog box. Click **OK**.

3. Open the Computer Management (Local) console tree. Click **Services and Applications>Services**. Start the **SNMP Service** to enable SNMP.

4. Double-click **SNMP Service** to open the SNMP Service Properties dialog box. Select the **Security** tab and specify an appropriate community.

5. To assign access rights to SNMP, Compuware recommends you select **Accept SNMP packets from these hosts** and set security to specific machines. In the text box, specify which machines you want to accept SNMP packets from. If you choose to expose SNMP to all external requests, select **Accept SNMP packets from any host**.

6. Click **OK**.

7. Open the **Windows Firewall Setting** dialog box. Select the **SNMP Service** check box. Click **OK**.

Windows Registry Remote Monitoring

In Vista, Registry remote monitoring is no longer available by default. To enable Windows Registry remote monitoring, you must start the Remote Registry service which now defaults to manual start up.

1. Open the Computer Management (Local) console tree. Click **Services and Applications>Services**.

2. Start the **Remote Registry** service.

WMI Remote Monitoring

Obtaining performance counters using WMI is the preferred way in Vista, due to security. To enable WMI remote monitoring, you must enable access to WMI information in the firewall.

1. Open the **Windows Firewall Setting** dialog box.

2. Select the **Windows Management Instrumentation (WMI)** check box. Click **OK**.

Running a Load Test and Analyzing Results

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 *QA*Load Script Development Workbench, *QA*Load 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 `\Middlewares\<middleware_name>\scripts` directory (for example, `\QAload\Middlewares\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 *QA*Load or create a *QA*Load 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 *QA*Load or to create *QA*Load 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.

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.

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.

Index