QlikView

QlikView Benchmark Test

A Hardware Benchmark Test

Published: October, 2013

www.qlikview.com

QlikView

Table of Contents

Introduction

Thank you for your interest in how QlikView performs on multiple hardware platforms. The QlikTech Scalability Center in Lund, Sweden has created the following test to help organizations like you to benchmark hardware to validate its performance against other similar environments. Your participation in this test is critical to gather information to facilitate recommendations for well performing hardware. By running the benchmarking package you will also verify that your configuration is optimal. Based on this goal, the following document will outline the requirements to configure and run this test in your environment. This test consists of two parts, stand-alone QV desktop based and QlikView Server Benchmark.

FTP Account details and files to download

Use an FTP client with SSL support of your choice is recommended. If you don't have access to any FTP client, you can access the ftp site over the following URL:
https://ftpeu.qliktech.com/thinclient/

- FTP Account Details
  - FTP (Site requires secure SSL connection): **ftpeu.qliktech.com**
  - User: **SC-files**
  - Password: **Files4Benchmarking**
- Files used during test session:
  - HWEtest_parallel_1G.qvw
  - HWEresults.xlsx
  - SC_BenchmarkApp1_200M.qvw
  - SC_BenchmarkApp1_5_scripts_QV10.zip
  - SC_BenchmarkApp1_5_scripts_QV11.zip

To perform the HW benchmarking test you need to download the HWE-files and the zip-file corresponding to the installed QVS version at the hardware subject to be tested (QV10 or QV11). Please note that HWEtest_parallel_1G.qvw is 7,6GB big and it might take long time to download. For your convenience there is folder named HWETest on FTP containing zipped version of this file divided in 100MB packages that can be downloaded instead (especially if using the browser thinclient). Download all zipped files and unpack the HWEtest_parallel_1G.qvw file.

Part 1 – Stand-alone tests

This tests measures QlikView performance during high CPU utilization. Test will take approximately 15 minutes to perform.

Pre-requisites - What You Will Need

- HWEtest_parallel_1G.qvw and HWEresults.xlsx downloaded from FTP server.
- Installed and licensed QlikView Desktop client.

Test execution

- Open HWEtest_parallel_1G.qvw in QlikView Desktop; wait for pop-up window to be shown.
- Write down values from pop-up window (Real seconds/CPU seconds) in the HWEresults.xlsx document)
- Close down QlikView desktop.

Repeat this scenario 5 times.

After test execution is done save HWEresults.xlsx file. Results for Stand-alone test should be submitted together with QlikView Server Benchmark results detailed information are presented in section 'Gather and Submit Results'

Part 2 - QlikView Server Benchmark

The second test is designed to test the performance of the QlikView server from a local client.

Pre-requisites - What You Will Need

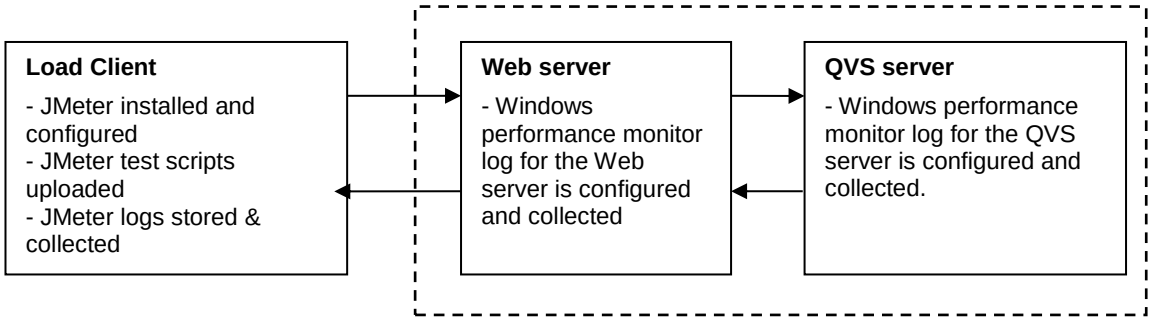
- Benchmarking package:
    - SC_BenchmarkApp1_5_scripts_QV10.zip or SC_BenchmarkApp1_5_scripts_QV11.zip
- Test application
    - SC_BenchmarkApp1_200M.qvw
- JMeter version 2.4 or newer
    - Download binaries available from here: http://jmeter.apache.org/download_jmeter.cgi
- Windows client pc with at least 8 GB of RAM
- Server running QlikView Server without any other resource consuming processes.

- Un-interrupted access to these computers for approximately four and a half hours

Test Environment Overview

The test needs to be initialized from a separate load client machine where the JMeter software and the benchmarking JMeter scripts are installed and configured correctly. Use a machine as load client physically positioned as close to the web server as possible to minimize impact of network latency etc. The load client machine does typically require at least 8 GB of RAM. During the test session it is important that both the client machine and QlikView server machine does not have any other resource consuming processes.

This benchmarking suite is designed for simulating the Ajax client against QlikView Server versions 10 and 11 (with either QlikView Web server or Microsoft IIS).



QlikView Benchmark Package

1.1 What is in the test package

The test package contains the following folder structure:

- Test scripts (where the test should be run)
  - batch (contains the batch scripts for the test)
  - benchmarkJMX (contains JMeter test scripts for the test, text files containing the randomization seeds and two text files used for header authentication)
  - logs (where JMeter runtime logs will be stored)
  - Results (will contain all the test results that you need to gather after the test)
  - BenchmarkingTestExecution.bat (test executable)
  - Config.txt (test configuration)
- Processes_template.xml (A template for the data collector set of the Windows performance monitor)

- ServerMetaFetcher.qvw (QV file which allow to gather target machine hardware information)

1.2 What should be done with package

Package should be unzipped to local drive on client machine, to folder of your choice.

Copy the “Processes_template.xml” file to a folder of your choice on the QVS server, and Web server if it’s a separate machine.

Copy ‘SC_BenchmarkApp1_200M.qvw’ downloaded from FTP to appropriate folder on QlikView Server machine and update document folders if necessary to grant access to the benchmarking application.

Prepare the Load Client machine

1.3 Load Client hardware requirements

It is advice to use a powerful machine as load client (i.e. small server). Running benchmarking package will consume significant amount of load client resources.

Tests done at Scalability Center prove that it is possible to run Benchmarking package from laptop. Machine used as load client should have 2 cores processor and 8 GB of RAM. We do not advise to use machines not meeting these requirements as load clients. It is also important to make sure that load client has at least 4GB of free RAM. It is advised to close other applications before running benchmarking package. First test has high demand if it comes to CPU usage.

It is required to use 64bit operating system and 64bit Java version (1.6.0_24 or newer).

1.4 Installing JMeter

JMeter is an open source, java based application used for load testing and performance measurements. For further information documentation, tutorials and source files can be found from the following URL http://jmeter.apache.org/.

Recommended version of JMeter is 2.4 or later which can be downloaded from: http://jmeter.apache.org/download_jmeter.cgi.

Once the binaries have been downloaded, unzip it to a folder in accordance to our recommendations below and it should be ready to run. No further installation needed.

1.4.1 User Privilege to run JMeter

It depends on the Operating System and the privilege setting of your user account. In some situation you will be asked to have the local administrator privilege to run JMeter under certain directory, and this will create issues if it was not setup properly.

We suggest two options that may help you to avoid the privilege issues:

1. Do NOT unzip the JMeter download package under “C:\”, “Program Files” or any system directory. Instead, extract it under a directory that you have the read/write/execute privileges to it.
2. Disable the UAC (User Account Control) settings in the JMeter load client machine. You can change it by clicking “Start” button →Type “UAC” input bar. Reduce it to the lowest level will disable the UAC. And you will be asked to restart your PC once you have disabled/enabled it.

1.5 Check java configuration and version

To be able to run the JMeter load generator you must make sure that there is a Java runtime environment installed at the client machine. 64bit JRE 6 update 24 or later is required.

Java JRE download are available on www.oracle.com page. Search for “Java SE downloads”, and choose Java SE Downloads link from results. It will redirect you to download page.

In order to be able to run JMeter java path needs to be set as Windows environmental variable. To check if it set or to set it up, navigate to Computer > System properties > Advanced system settings> Environment Variables. Find variable PATH and check if variable value contains path to the bin folder of your JRE installation (e.g.: C:\Program Files\Java\jre6\bin if JAVA 6 was installed in default location). If you can’t find JAVA path add your JAVA path to variable values.

For further details about how to set JAVA path refer to:
http://www.java.com/en/download/help/path.xml

To verify path settings and java version open command line tool from start menu and type:
java -version

Above there is example output. Make sure that you’re running 64-Bit version.

Try to start JMeter using jmeter.bat file, which can be found in JMeter folder in bin folder.

If JMeter does not launch, make sure that you run it as administrator (right click select run as Administrator). In case administrator privileges are needed, attached batch file for running scripts in non-GUI mode (BenchmarkingTestExecution.bat) should be run with using ‘as Administrator’.

1.6 Recommendations for BIOS settings

Please review the following document http://community.qlikview.com/docs/DOC-2362 on Community pages. It should provide a starting point on which BIOS settings that are experienced as beneficial for QlikView.

Configure the Performance Data Collector

Note: The process described in this instruction is ONLY valid with Windows 7/Windows Server 2008 R2/Windows Vista operating system. For Windows Server 2003 and other OS, you need to find other ways to log the following performance counters of all processes:

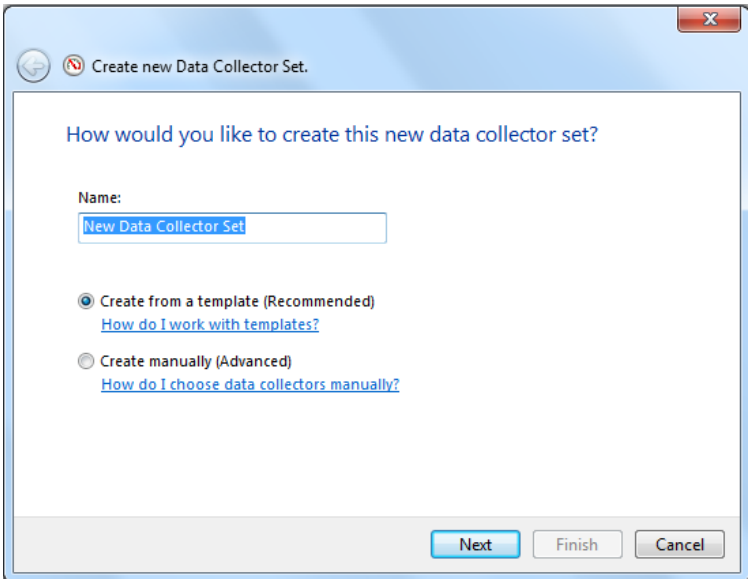
- "% Privileged time"
- "% Processer time"
- "Pool Nonpaged Bytes"
- "Pool Paged Bytes"
- "Private Bytes"
- "Virtual Bytes"
- "Working Set"

A Data Collector Set is the building block of performance monitoring and reporting in Windows Performance Monitor. It organizes multiple data collection points into a single component that can be used to review or log performance. In order to collect the system performance information from Windows server, you need to configure the performance data collector on both QVS server and the Web server machines (if they are on the same server then you just need to configure it once).

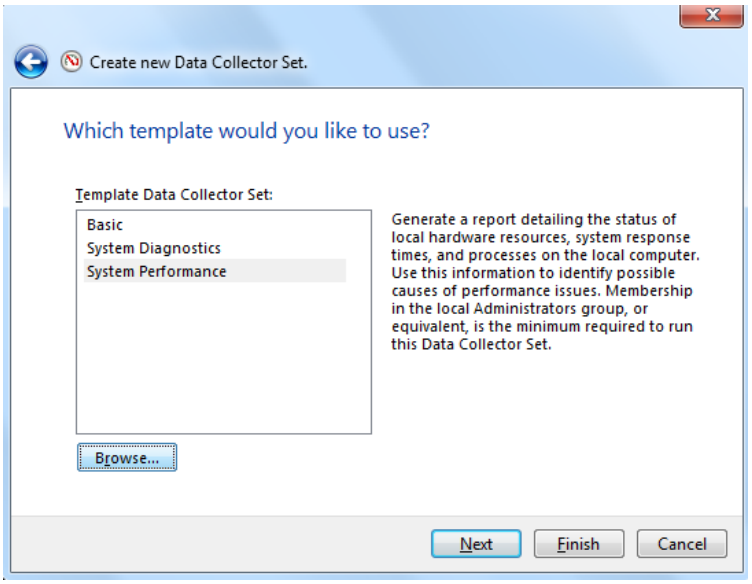
First, copy the "Processes_template.xml" file from the extracted test package to the folder of your choice on QVS and Web server machines.

Start the "Server Manager" of the Windows server, and go to "Diagnostics" > "Performance" > "Data Collector Sets" > "User Defined". Add a new DataCollector set to be like the example below.

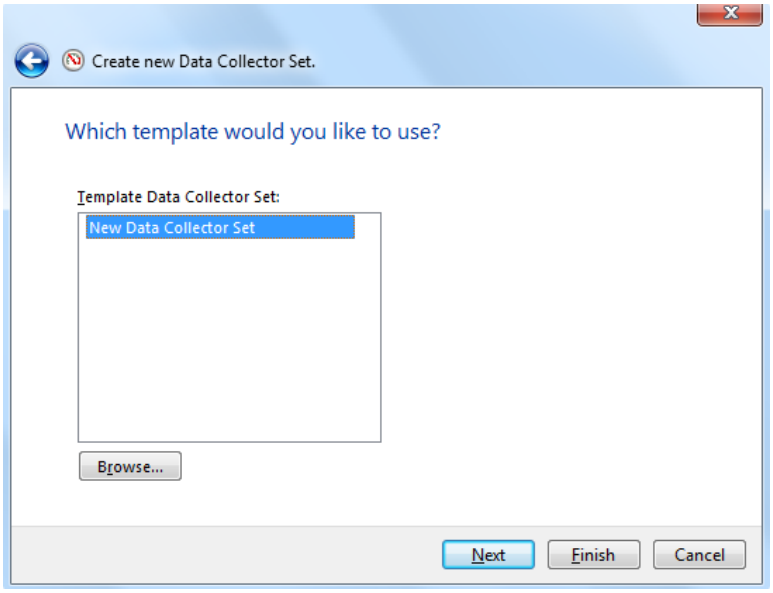
Right click and select "New > Data collector set", the following window will appear:



Select “Create from a template” and click Next.



Click “Browse” and select the “Processes_template.xml” copied from the benchmark test package.

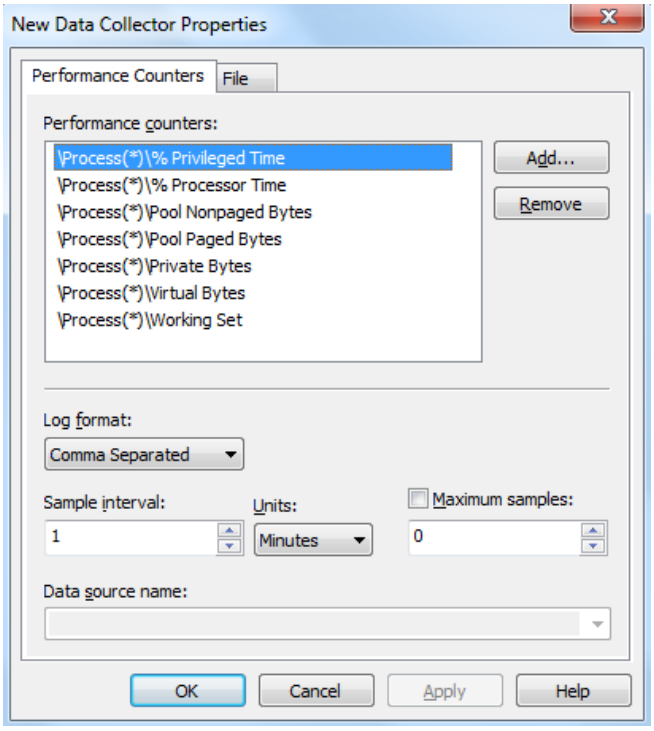


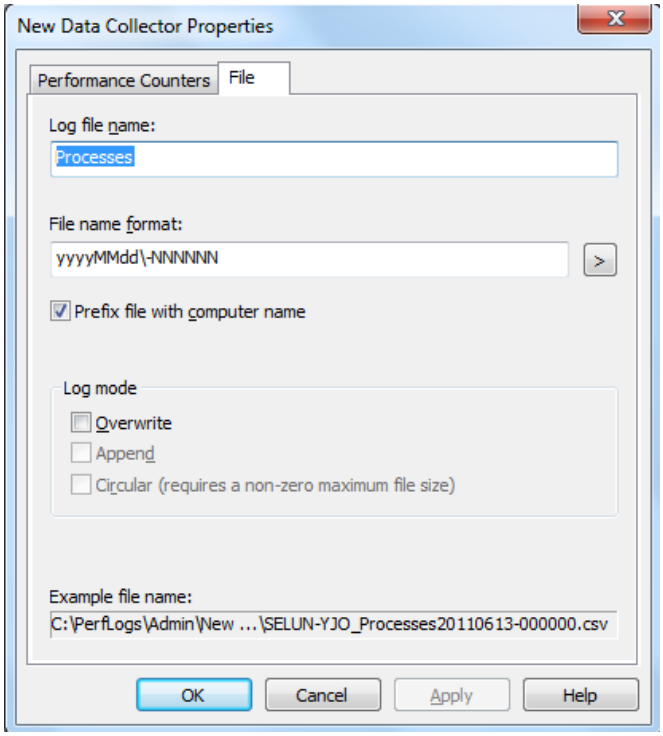
Click “Finish” to complete the setting for the data collector set.

For more detail information about creating a DataCollector set, please refer to:

http://technet.microsoft.com/en-us/library/cc722148.aspx

Below is an example of well created data collector set from the template.



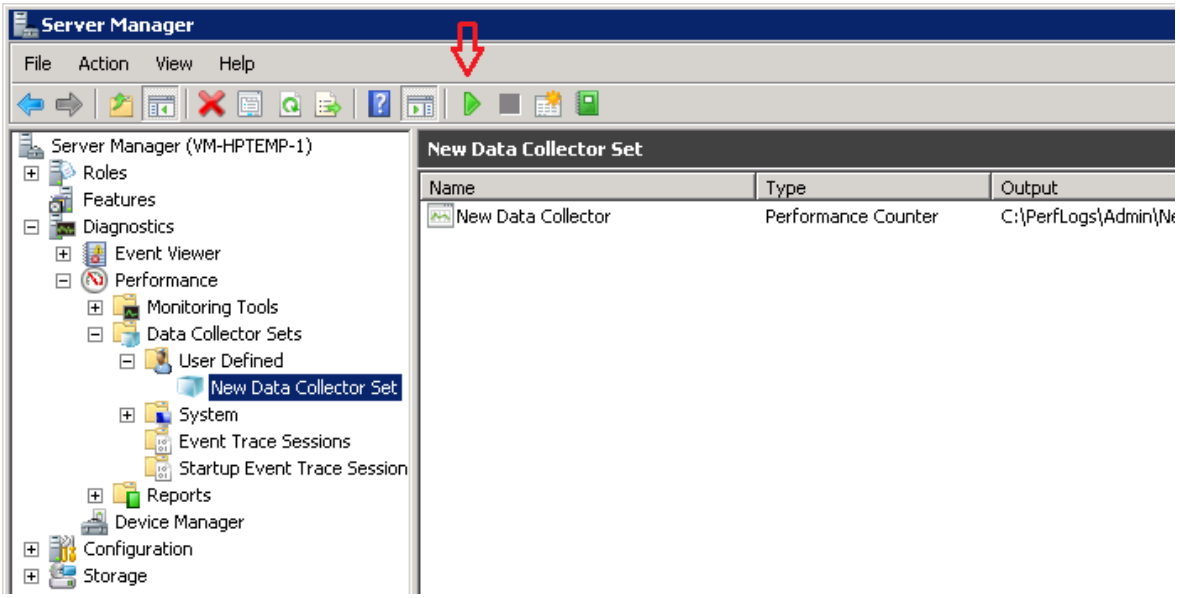


After the DataCollector set is created, make sure that the processes subject for monitoring are running (e.g. QVS and Web Server (qlikviewwebserver for QV Web server or w3wp for IIS) processes can be seen in task manager when starting DataCollector).

Note: If IIS is used as Web Server access QlikView from client machine and open any document, before starting performance counters – otherwise Web Server process might not be logged.

Note: Every time after you have rebooted the machine, you need to start the DataCollector set from the server manager manually (and make sure QVS and IIS is running each time DataCollector set is restarted).

To start the data collector select it and click “Start” icon in taskbar.



Performance Data Collector GUI in Windows Server Manager

1.7 Administrator privilege on server

Make sure that user account used on client machine is part of Administrators group on the server. This is needed to perform server restart which is automatically done before each test is started.

1.8 QlikView Server and Web Server settings in QMC

Before you modify settings make sure application 'SC_BenchmarkApp1_200M.qvw' was copied to QlikView Server machine and if needed Document Folders were updated (System > Setup > QlikView Servers > Folders).

In order to use Benchmarking package with provided QlikView application containing embedded document license changes in QMC needs to be done.

1.8.1 Session recovery

This setting needs to be changed because JMeter simulates many sessions by reusing the same users. Recovering from last state where certain users were active might cause issues for the predefined script.

.1.8.1.1 For QlikView version 10

“Prohibit Session Recovery” must be turned on (selected checkbox)

The setting “Prohibit Session Recovery” can be found in the QlikView Enterprise Management Console (QEMC) “System > QlikView Servers >select the host of the QlikView server > Documents” tab.

Make sure that Prohibit Session Recovery is selected. Apply changes.

.1.8.1.2 For QlikView version 11

“Allow Session Recovery” – should be turned off (deselected checkbox)

The setting “Allow Session Recovery” can be found in the QlikView Management Console (QMC) “System > QlikView Servers > select the host of the QlikView server > Documents” tab.

Make sure that Allow Session Recovery is deselected. Apply changes.

1.8.2 Logging

In order to get more detailed information for analysis it is advices to change logging level. In order to do so, go to System > QlikView Servers > select the host of the QlikView server > Logging

Make sure that in Logging section values: “Enable Session Logging” and “Enable Event Logging” are selected. In Event log Verbosity section choose High.

Because event logs will grow over time using Split Files option might be a good idea.

Apply changes.

1.8.3 Security

For script to be able to use document with the embedded license, QV server should be in DMS mode. To change this setting go to System > QlikView Servers > select the host of the QlikView server > Security

In Authorization section select “DMS authorization (QlilkView controls file access)”. Apply changes.

1.8.4 Document settings

In order to be able to use embedded document license, following changes need to be done on document level: set users authorized to access document and enable dynamic document CALs assignment.

In QEMC (ver. 10)/QMC (ver. 11) go to: Documents > User documents > Select server > Select SC_BenchmarApp1.qvw

Select tab Document CALs, in section Document CALs there should be information: 1000 Embedded Document CALs. Make sure that checkbox “Allow dynamic CAL assignment” is selected. Apply changes.

In some rare cases for some QlikView versions the Document CALs tab might not be present. In such case stopping QlikView Server service and starting it again should solve the issue.

.1.8.4.1 For QlikView version 10

Select tab Authorization, use green plus icon next to “Recipients”, new entry will be shown. Click on ‘Edit’ (pencil) icon. In new window click on ‘Users’ icon and select “All Authenticated Users”, confirm your selection with ‘OK’ button. Apply changes.

.1.8.4.2 For QlikView version 11

Select tab Authorization, use green plus icon next to “User Authorized to Access Document”, new entry will be shown. Make sure that Access is set to Always and User Type to “All Authenticated Users”. Apply changes.

After changes describe above were done on server. Open the Access Point using client machine and access SC_BenchmarkApp1. Copy the URL to document and save it. It will be needed later on for test configuration.

1.8.5 Enable header authentication

Go to System > Setup > QlikView Web Servers > select available server > select tab Authentication

Set Authentication Type to Header and apply changes.

Below in section Parameters value Header name is specified (default value is: QVUSER), if this value is changed to something else this value needs to be updated in the package config.txt file, described later.

1.9 QVWS settings

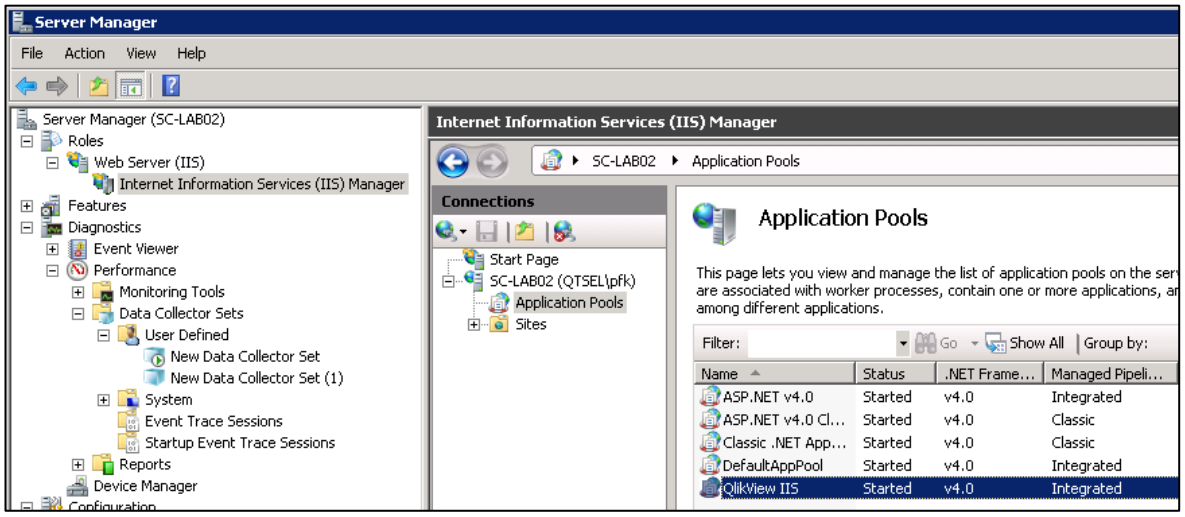
No changes are needed if QVWS is used.

1.10 IIS settings

If IIS web server is used we recommend changing settings according to instructions below.

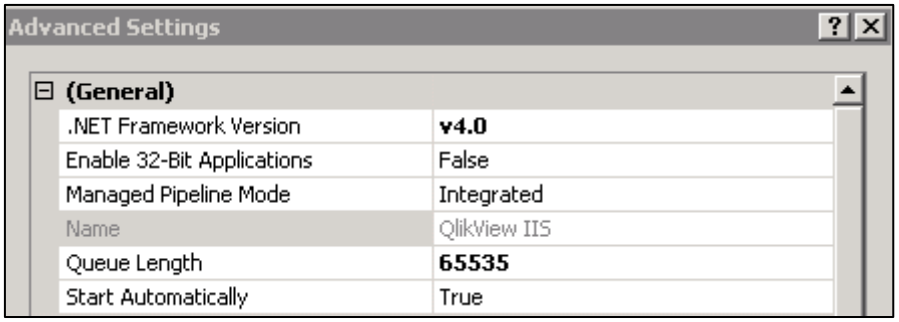
Start Server Manager, in left pane select Roles > Web Server (IIS) > Internet Information Services (IIS) Manager

In section Connections select machine name, unfold this section and select “Application Pools”

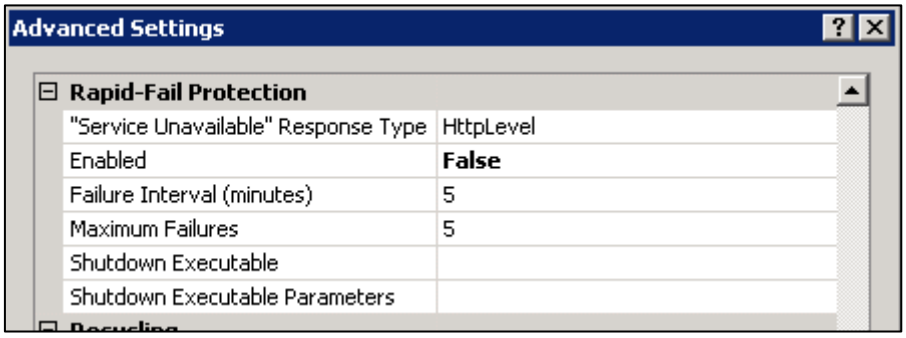


Find QlikView IIS application pool select it, right click and choose Advances Settings.

Change Queue Length in (General) settings to 65535



Scroll down to Rapid-Fail Protection and change Enabled to False



Confirm your changes with “OK” button.

Setting up the Benchmarking Test Package

1.11 Edit the configuration

Find the ‘config.txt’ file under the folder ‘Test scripts’, and open it with your favorite text editor.

The ‘config.txt’ file contains the following configurations:

JMeter Installation Path =

Header Name =

Document URL =

QVS Hostname =

Web Server Hostname =

Web Server Type (IIS/QVWS) =

QVS Version =

1. The first step is to tell the test package where to find the ‘jmeter.bat’ file in order to start JMeter. Copy the full path of your JMeter installation and paste it after the “JMeter Installation Path =”. For example:
   - JMeter Installation Path = C:\Jmeter\jakarta-jmeter-2.4\bin
2. In order to allow proper header authentication, header name needs to be specified. Header name is set to default QVS value. If you haven't change header name in QVS no changes are needed. To check Header name in QMC go to System > Setup > QlikView Web Servers > select available server > select tab Authentication.

Authentication Type should be set to Header, below in section Parameters value Header name is specified.

- Header Name = QVUSER

3. Paste document URL which you have copied during server setup (described in section 1.8.4 in this document) into the 'config.txt' file after "Document URL =". For example:

   - Document URL = http://target.server.com/QvAJAXZfc/opendoc.htm?document=SC_BenchmarkApp1_200M.qvw&host=Local

   Note: Path to document should not contain special chars including space character or its URL encoded representation %20.

4. Provide hostname of the QVS machine and Web Server machine. Hostname is computer name – to check it start command line and type hostname. Type the same hostname if QVS and Web Server are installed on the same host. For example:

   - QVS Hostname = QVS-HOST
   - Web Server Hostname = WS-HOST

5. Input the type of the Web Server you are running. There are ONLY two web server types supported: QlikView Web Server (QVWS), and Microsoft Internet Information Server (IIS). Choose either "QVWS" or "IIS" to type in.

   - For example: Web Server Type (IIS/QVWS) = IIS

6. It is recommended to note down the QlikView Server version that has been tested. Put the version information after "QVS Version =" and it will be logged in the test log names. For example:

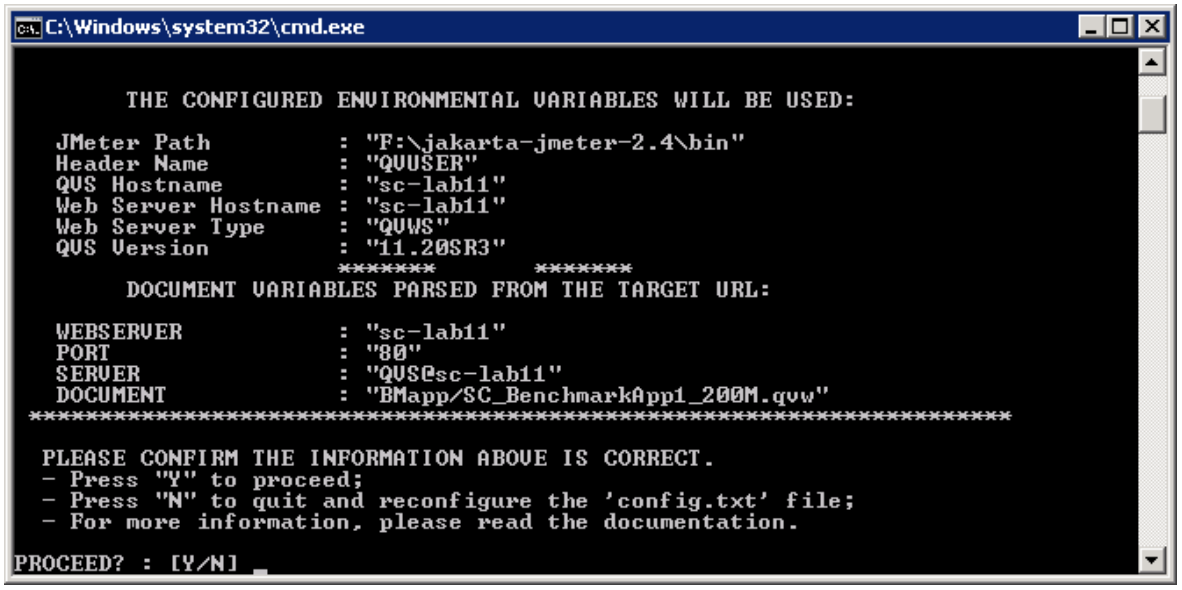
   - QVS Version = 11IR

Save changes done in config.txt file.

Test Execution

You can start the automated test suite by clicking and running the "BenchmarkingTestExecution.bat" file in the 'Test script' folder, under the folder where you extracted the test package.

First ping test will be done for QlikView Server machine (and Web Server if it is placed on another machine) after it is done following command line window will be shown:



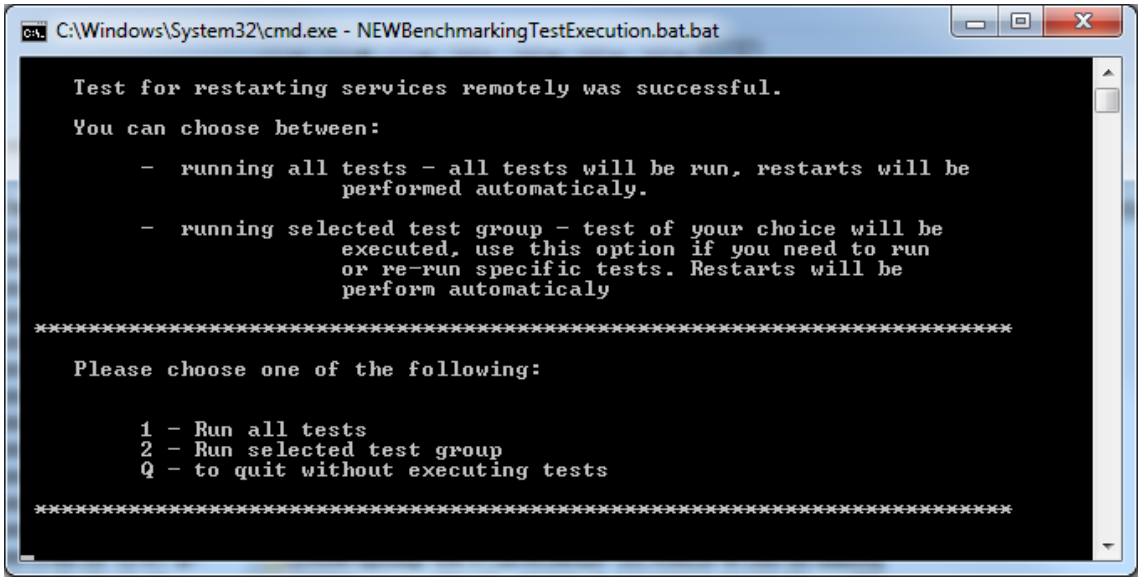
The purpose is to be able to check information provided in config.txt file before running test.

After confirmation is done check will be performed to determine if automatic server restart is possible. It is mandatory to restart QVS and WebServer before each test.

Depending on result different path of execution can be chosen.

Case 1: Automatic restart is possible

Information text will be displayed:



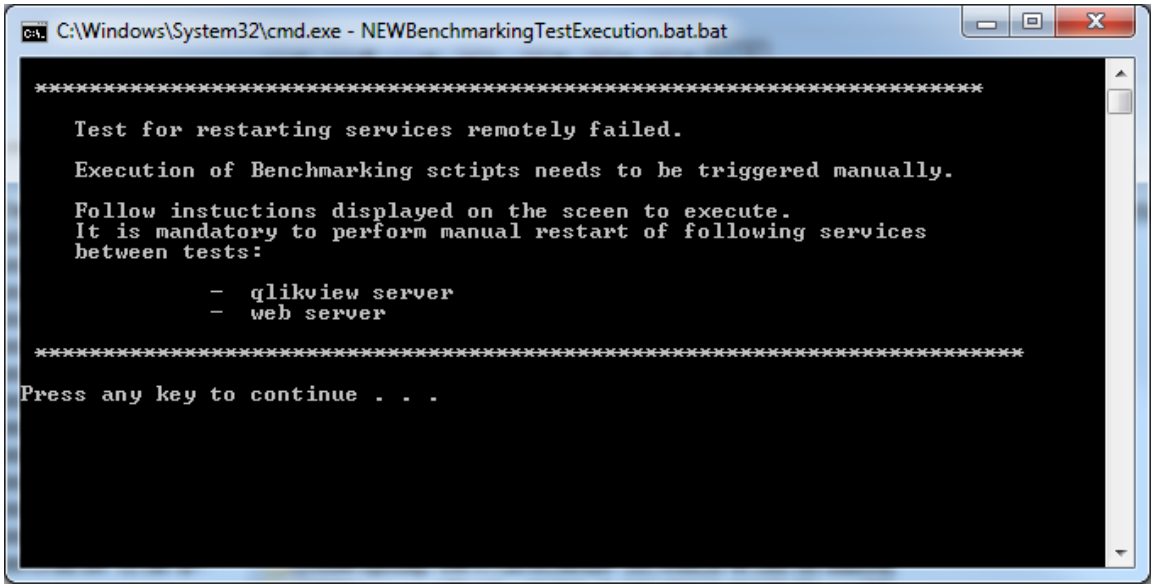
1 – Run all tests – will start execution of all tests scenarios which will take approximately 4,5h

2 – Run selected test group – if there is a need to run only selected test groups this option should be used. QVS and Web Server will be restarted before and after test.

Q – quit without executing tests - will end the execution.

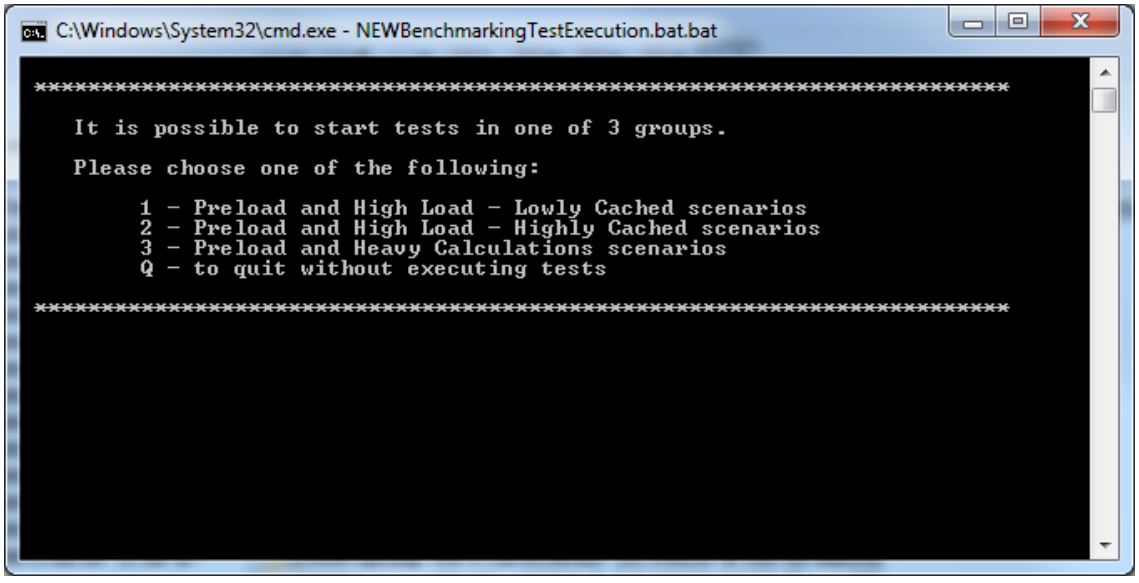
Case 2: Automatic restart is not possible

Information text will be displayed:



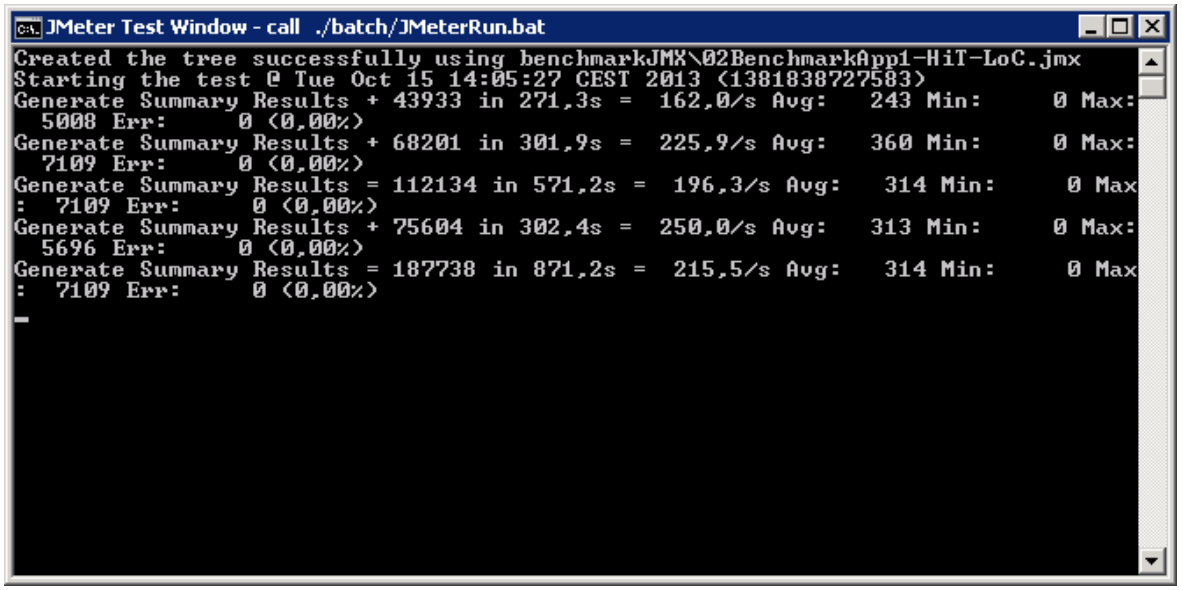
Please note: QVS and Web Server have to be manually restarted between running different test groups.

Next screen contains names of 3 groups of the test and choice to run one of the groups or quit without executing tests.



The tests will be automatically executed one by one in a pop-up window. Each test run consists of preloading document part and test part. If preload is not successful error information should be displayed. It is advised to monitor test execution window at the beginning of the test and from time to time during test execution. Thanks to this it can be seen if test execution is successful from JMeter perspective.

Example output is presented below:



New summary row will be printed approximately every 5 minutes. Value which should be monitored is ‘Err’ in rows containing equal sign (=). Growing error rate suggests that test execution is not successful.

Let the test script run until it finishes. However in case you want to stop the test before it ends, you can do it with sending “Ctrl + C” in **both windows**.

Restore Configuration

1. Erase benchmarking application from the QVS
2. Remove document folder to benchmarking application if added
3. Set “Allow Session Recovery” ver. 11 / “Prohibit Session recovery” ver. 10 to setting from before you start the test
4. Change logging level to settings from before test
5. Stop monitoring DataCollector set

Gather and Submit Results

Create folder with your name/your company name. All data needed for test analysis should be copied to this folder. Copy HWEresults.xlsx file containing results from stand-alone tests.

Copy ‘Results’ and ‘logs’ folders from ‘Test scripts’ folder in Benchmarking package to folder you have created.

From benchmarking package folder open the ServerMetaFetcher.qvw and enable Macro. Add all IP addresses or machine names, for all machines used in the testing, to the table. Click the "Fetch Properties"-button. Check the details table to ensure that everything is collected.

Make sure that the WorkingSet-limits should be set according to the settings in System → Setup→Performance for the machines, as these are defaulted to min 70 and max 90 and not read from the server.

Click the "Export ServerInfo"-button. This will save an xml file, called ServerInfo.xml. Copy this file to folder you have created.

Copy Performance Counters from QV Server machine (and Web Server if installed on separate machine) to folder you have created.

Go to the directory **“C:\PerfLogs\Admin\”***Your DataCollector set name’*’**\”** and fetch logs that has been recorded during the test execution.

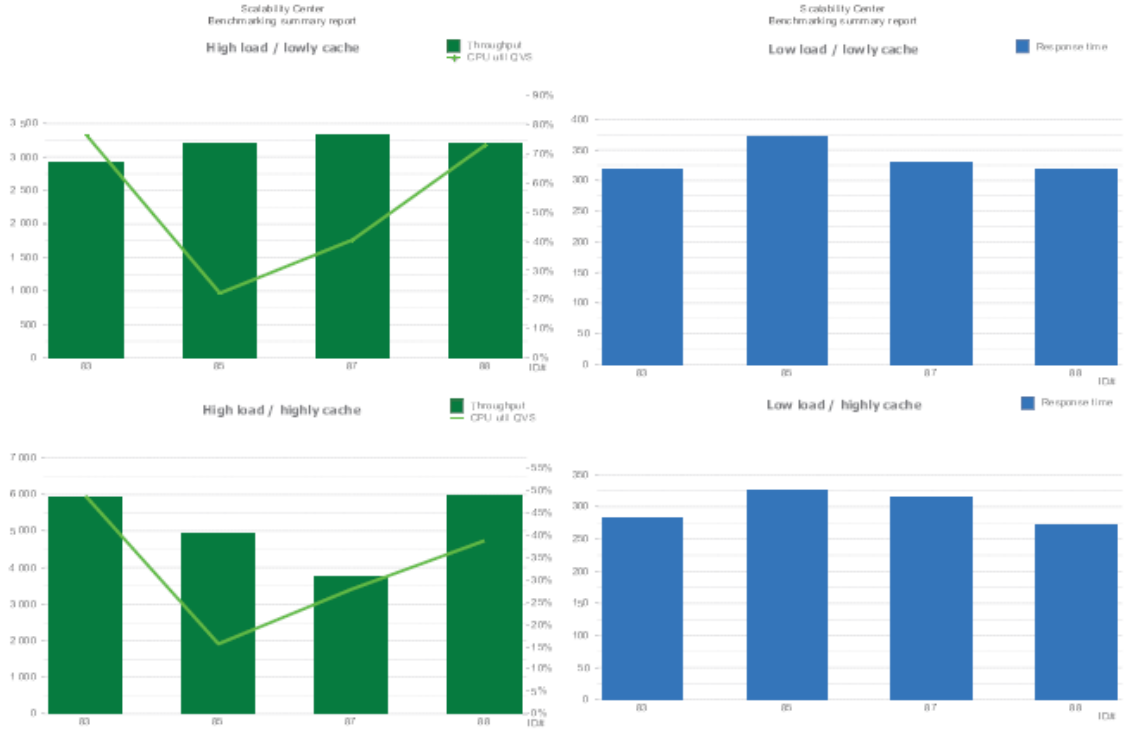
Collect QVS session log and QVS event log from QlikView Server machine. By default logs are placed in: C:\ProgramData\QlikTech\QlikViewServer. Place these logs into folder you have created.

Zip folder in which you placed all data for result analysis, and upload the results to TestResults folder on FTP server from which you have downloaded benchmarking tests. FTP server information can be found in section FTP Account details and files to download in this document.

Please also send an email to ScalabilityLab@qlikview.com notifying that there are new files uploaded.

What You Will Receive Back

By participating and providing results you will receive feedback containing evaluation of your server performance in comparison with similar machine. Summary of servers’ performance and brief discussion over results will be presented. Example of comparison data might look similar to this:



Legend

ID#	CPUs	Cores per CPU	Logical cores	Clock speed	RAM (GB)	HT	NUMA
83	2	6	12	3,466 GHz	96	FALSE	Disabled
85	8	10	80	2,4 GHz	512	FALSE	Disabled
87	4	10	40	2,4 GHz	256	FALSE	Disabled
88	2	8	16	2,9 GHz	128	FALSE	Disabled

ID#	scenario	CPU Total	CPU IIS	CPU QVS	Avg per action	Avg over time	#Req/min
83	High CPU Load, Lowly Cached	77,8%	-	76,2%	998	987	2 906,4
	High CPU Load, Highly Cached	51,9%	-	48,7%	496	493	5 900,5
	Low CPU Load, Lowly Cached	0,7%	-	0,7%	316	318	11,0
	Low CPU Load, Highly Cached	0,2%	-	0,2%	283	284	10,9
	Heavy calculations	63,1%	-	63,5%	6 464	7 385	33,9
85	High CPU Load, Lowly Cached	22,9%	0,0%	22,2%	913	936	3 201,3
	High CPU Load, Highly Cached	15,9%	0,0%	15,6%	590	587	4 919,4
	Low CPU Load, Lowly Cached	-0,9%	0,0%	0,2%	370	397	10,5
	Low CPU Load, Highly Cached	0,1%	0,0%	0,1%	326	337	10,6
	Heavy calculations	25,9%	0,0%	25,7%	2 921	2 883	42,7
87	High CPU Load, Lowly Cached	41,0%	-	40,5%	883	879	3 313,5
	High CPU Load, Highly Cached	28,5%	-	27,8%	769	764	3 770,8
	Low CPU Load, Lowly Cached	0,8%	-	0,7%	330	335	10,5
	Low CPU Load, Highly Cached	-0,2%	-	0,1%	314	315	10,5
	Heavy calculations	32,7%	-	33,5%	1 844	1 836	46,3
88	High CPU Load, Lowly Cached	74,3%	-	73,1%	918	915	3 186,1
	High CPU Load, Highly Cached	40,8%	-	38,7%	487	482	5 955,9
	Low CPU Load, Lowly Cached	0,7%	-	0,6%	319	322	10,9
	Low CPU Load, Highly Cached	-0,4%	-	0,2%	271	271	10,7
	Heavy calculations	59,7%	-	60,9%	6 962	7 792	32,8

In addition to the feedback provided from the Scalability Center one can by advantage perform analytics of the benchmarking results by oneself. For instant feedback you can use Analyzer part of the QVScriptGenTool available on Community pages: http://community.qlikview.com/docs/DOC-2705.

It is recommended to run Benchmarking Package with different BIOS settings and QV Server configurations to verify optimal performance.