QlikView

QLIKVIEW SERVER LINEAR SCALING

QlikView Scalability Center Technical Brief Series

June 2012

qlikview.com

Introduction

This technical brief presents an investigation about how QlikView Server scales in performance when serving multiple QlikView applications. One of the major QlikView differentiators is our customers' Business Discovery adoption path. Typically, QlikView enters enterprises with a couple of QlikView applications that are created to solve a workgroup's business problems. In weeks, as the business users recognize the ease of use and the self service nature of QlikView, there is an exponential increase in the number of QlikView applications created.

From an IT department's perspective, it is very important to have a predictable and performing Business Discovery infrastructure in these highly growing environments. It's important to understand that QlikView's performance scales uniformly as new QlikView applications are created.

The aim of the investigation explained in this paper is to observe the behavior and effects from having multiple QlikView applications loaded and accessed at the same time on the same server and compare the result with testing the same applications loaded and accessed in sequence.

As the Business Discovery environments have high adoption rates, QlikTech recognizes the importance of scalability and predictability of the QlikView architecture.

The intention of this paper is to share the test results conducted by the QlikTech Scalability Center on linear scalability of the QlikView Server. These results should be taken as guidance. It is recommended to review the QlikView Architecture and System Resource Usage Technical Brief in order to get a fundamental understanding of the various QlikView components and how they utilize various hardware resources such as RAM and CPU.

On the remainder of this paper, the test methodology and configurations are explained and findings are summarized with explanations of the observed performance.

QlikView Resource Need Scales With the Number of Loaded QlikView Applications

The aim of the investigation is to observe the behavior and effects from having multiple QlikView applications loaded and accessed on the same server and compare the result with testing the same applications in sequence in the same environment. The ideal result of the investigation would be that adding up the outcome of the individual application loads matches the outcome of the applications loaded in parallel. The investigation will show the actual results from the perspectives of CPU usage, RAM usage, throughput and average response times. The result from this investigation should serve as evidence on how QlikView Server scales for multiple applications running in parallel.

Technical Information Regarding the Tests

Five QlikView applications were used throughout this investigation ("App1.qvw", "App2.qvw", "App3.qvw", "App4.qvw", and "App5.qvw".) The applications are categorized as small, medium and large size applications (see Figure 1).

The tests are carried out in two groups. In the first group two small-size QlikView applications are used to test the RAM and CPU utilization of the QlikView Server loading equal-size and light-weight applications. In the second group, four QlikView applications, ranging from small size to large size, are used to observe the resource utilization on the QlikView Server. Finally, the test results of the two groups are compared.

Figure 1. The technical information of the test environment

Environment	Attribute	Value
QlikView Applications	Small size applications	App1.qvw: 79K (# of records)
	Middle size application	App2.qvw: 6.7M App3.qvw: 56.5M App4.qvw: 603M
	Large size applications	App5.qvw: 400M
IIS	Version	7
QlikView Server	Version	v10. Service Release 2
Small Server	RAM	92 GB
(SC-MEDIUM)	CPU	12 cores, 3.33GHz
Large Server	RAM	256 GB
(SC-LARGE)	CPU	32 cores, 2.27GHz
Load Client Machine	Load generator	JMeter 2.4
	RAM	32 GB
	CPU	8 cores

Test Methodology

For this scalability testing, JMeter, (a load/performance testing tool), is used to simulate the user interaction scenarios with five different QlikView applications. A link to JMeter documentation is available in the references section. A specific JMeter script is created for each QlikView application simulating a predefined scenario. The scripts are then run against one application at a time (sequence test) and against all applications at once when performing parallel test executions (parallel test).

In the first group of the test, "App1.qvw" and "App2.qvw" are tested against the smaller server referred to as SC-MEDIUM. In the second group of test, "App1.qvw", "App3.qvw", "App4.qvw", and "App5.qvw"are tested against the bigger server referred to as SC-LARGE.

The test scenarios of loading the QlikView applications are tuned to avoid saturating the CPU and RAM on the target QlikView Server. The reason for this is, as at point of saturation, it will not be visible how much RAM that is actually required per application. A sample user interaction script scenario is provided below.

Figure 2. A sample user interaction script used during the testing

Step	Description
1	Login to the server's Access point
2	Open the QlikView application (App1.qvw)
3	Change the active sheet, static selection
4	Chart selection, zoom random area
5	Listbox selection (LB1), random selection
6	Listbox selection (LB2), random selection
7	Listbox selection (LB3), random selection
8	Clear all selections, static selection

Test Results

1. GROUP 1 TESTS: LOADING TWO QLIKVIEW APPLICATIONS

This test result shows the effect of loading two QlikView applications individually versus loading them in parallel on the QlikView Server.

Figure 3. Group 1 test results (CPU and Private Byte)

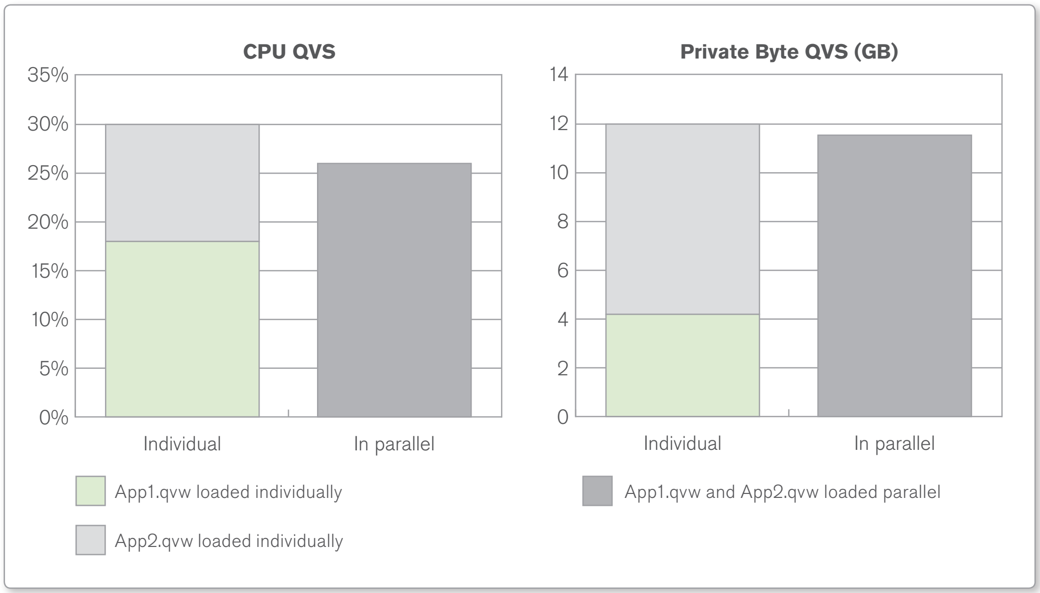
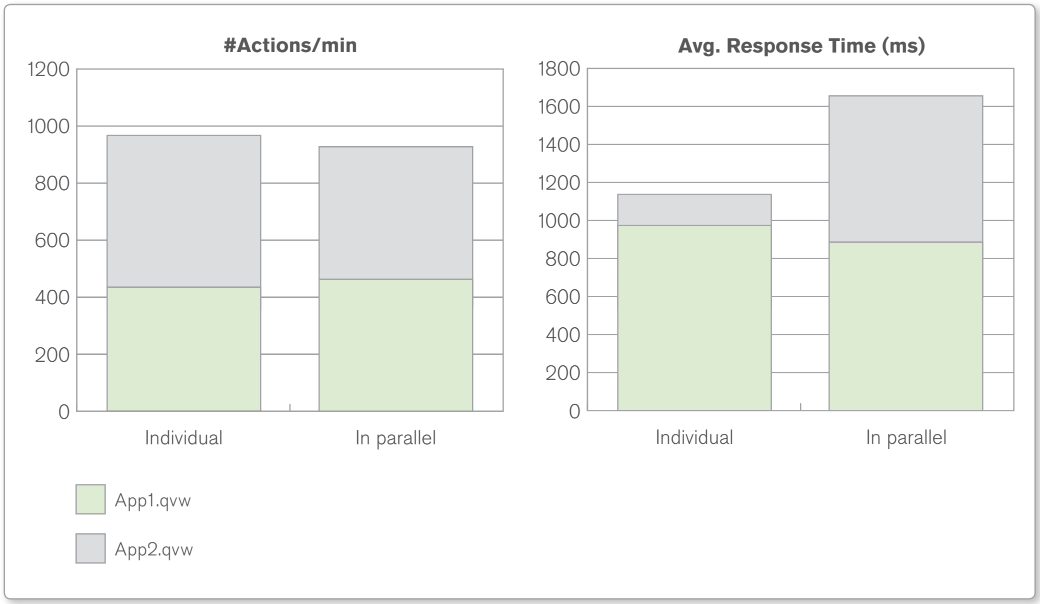


Figure 4. Group 1 test results (Number of Actions per minute and Average response time in ms)



2. GROUP 2 TESTS: LOADING FOUR QLIKVIEW APPLICATIONS

This test result shows the effect of loading four QlikView applications individually versus loading them in parallel on the QlikView Server.

Figure 5. Group 2 test results (CPU and Private Byte)

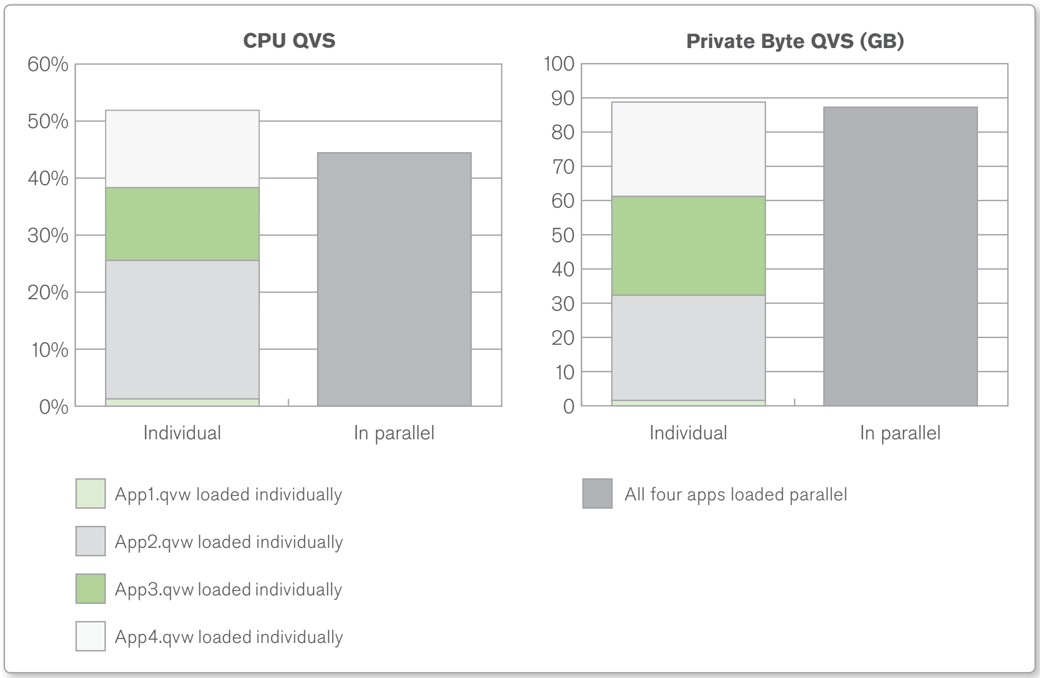
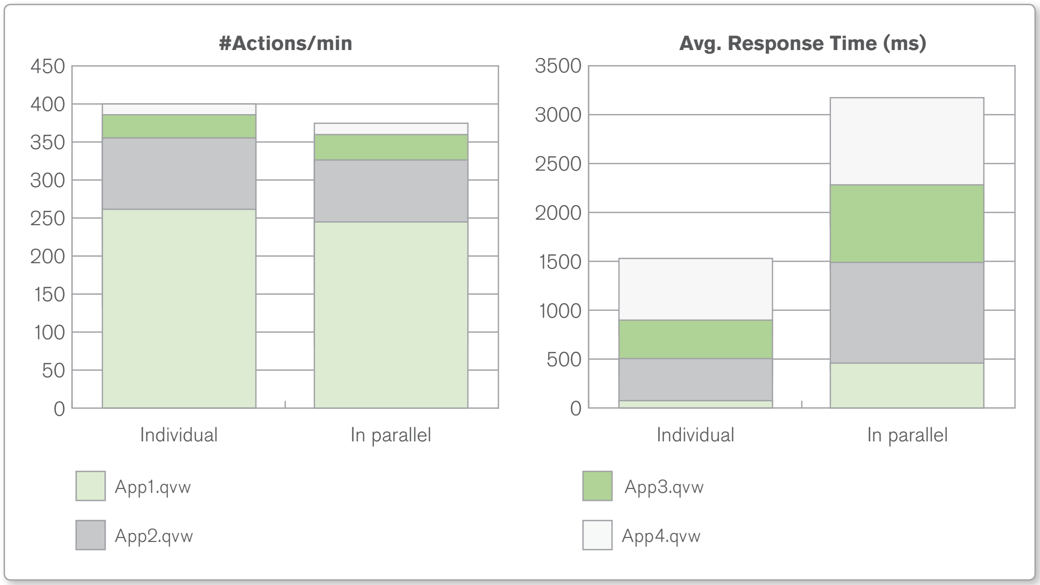


Figure 6. Group 2 test results (Number of Actions per minute and Average response time in ms)



Summary

The test result shows that by adding up the resource usage of individual QlikView applications, it is possible to get a close approximation from a resources perspective (i.e. CPU, memory) of what resources are needed when the QlikView applications run in parallel. The test results also show that the throughputs are almost at the same level when loading the QlikView applications in sequence and in parallel, due to the fact that CPU did not saturate during the test scenarois. The only thing to notice is that the average response time increases for the parallel load. This means that the system is capable of serving all incoming requests during the parallel load (i.e. not saturating in CPU), but as each request for processing is competing with requests against other applications, additional time is spent for queuing resulting longer response times.

The test results show the same pattern for the smaller server (SC-MEDIUM) with fewer cores and memory as for the larger server (SC-LARGE) with more cores and memory.

These observations proved that QlikView's performance scales uniformly as new QlikView applications are added to the QlikView Server. It is viable to assume that QlikView Server consumes approximately the same amount of resource when QlikView applications are loaded and accessed in parallel versus individually. This provides predictability for the QlikView administrators managing QlikView applications at a shared environment. By looking at individual QlikView application's resource usage, they can get an estimate on how much resource will be required in total.

References

QlikView Development and Deployment Architecture Technical Brief
www.qlikview.com/…/global-us/direct/datasheets/DS-Technical-Brief-Dev-and-Deploy-EN.ashx

QlikView Architecture and System Resource Usage Technical Brief
www.qlikview.com/…/DS-Technical-Brief-QlikView-Architecture-and-System-Resource-Usage-EN.ashx

QlikView Scalability Overview Technology White Paper
http://www.qlikview.com/us/explore/resources/whitepapers/qlikview-scalability-overview

Tool to create load/performance tests of QlikView 10 and QlikView 11
http://community.qlikview.com/docs/DOC-2705