QlikView
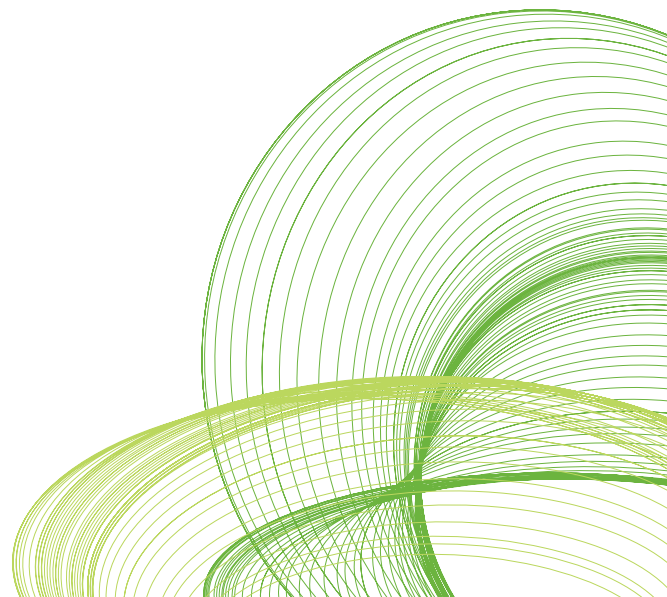
IMPACT OF NUMA ON QLIKVIEW 11 PERFORMANCE

A Scalability Center Technology Brief

September 2013
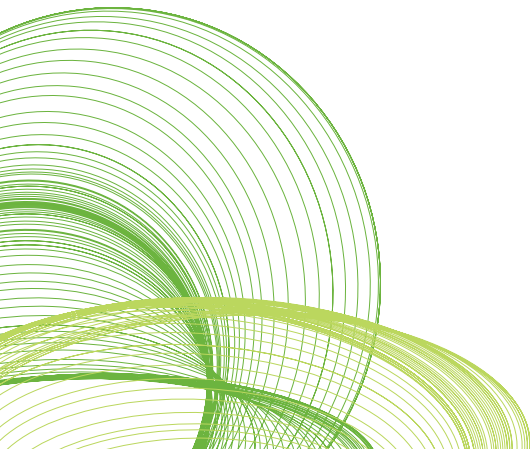
qlikview.com

QlikView

Table of Contents

Introduction and Assumptions

This document briefly introduces a common computer memory design known as Non-Uniform Memory Access (NUMA) and reports findings of its impact on performance in a QlikView 11 environment. The information outlined in this document is most beneficial to those in IT-related functions and those responsible for the total health and scalability of a QlikView deployment. This document assumes that QlikView is the only software application running on the server (recommended) and does not make any assumptions or recommendations for other software that may take advantage of NUMA. The reader should be familiar with QlikView and the hardware (BIOS settings) that QlikView is installed on and should have the authority to make the appropriate changes if deemed necessary.

What is Non-Uniform Memory Access (NUMA)

Multiprocessing involves using more than one CPU for speedy computation. With today's hardware variations, there are a variety of ways to implement it. NUMA is a multiprocessing memory architecture designed to surpass the scalability limits of other known architectures. It allows a processor to access its own local memory much faster than non-local memory (memory local to another processor or memory shared remotely between them). Modern CPUs work considerably faster than the main memory they use, causing a bottleneck as the CPU waits for data from its memory. Multi-processor systems without NUMA can make the problem even worse. NUMA attempts to address this problem by providing separate memory for each processor (a NUMA node), avoiding the performance hit when several processors attempt to address the same memory.

QlikView and NUMA

NUMA technology typically provides performance gains over other multiprocessing architectures especially if a software application is designed to take advantage of NUMA. However, extensive tests at our Scalability Center prove that NUMA-enabled servers have a negative impact on QlikView Server performance. This impact is less noticeable on systems with fewer CPU sockets as the need to access remote memory is reduced. Therefore, for NUMA architectures, it is beneficial to disable NUMA in the BIOS settings of the hardware running the QlikView Server in order to achieve optimum performance. Some vendors have removed the ability to turn off NUMA in the BIOS. If that is the case, a soft-switch has been implemented into the QlikView configuration which circumvents NUMA memory allocation. Since QlikView 11.2 SR2, QlikView will automatically detect if NUMA has been disabled and adapts appropriately for optimal performance. (For QlikView releases prior to QV11.2 SR1, see this document for more information on this setting: http://community.qlikview.com/docs/DOC-2362)

If a QlikView application (.QVW) is opened and loaded into memory on a NUMA-enabled system with the soft-switch logic disabled, the operating system will use the local memory for one CPU at a time. When necessary, it will use remote memory within another NUMA node (See Figure 1). (CPU access time to its local memory is much faster than access to remote memory). QlikView scales well over multiple cores resulting in those cores needing access to the same memory if "everything" is stored in one place. Since QlikView is not optimized for NUMA, please take note of the soft-switch setting for QlikView versions prior to 11.2 SR2 or disable NUMA in the BIOS if applicable.
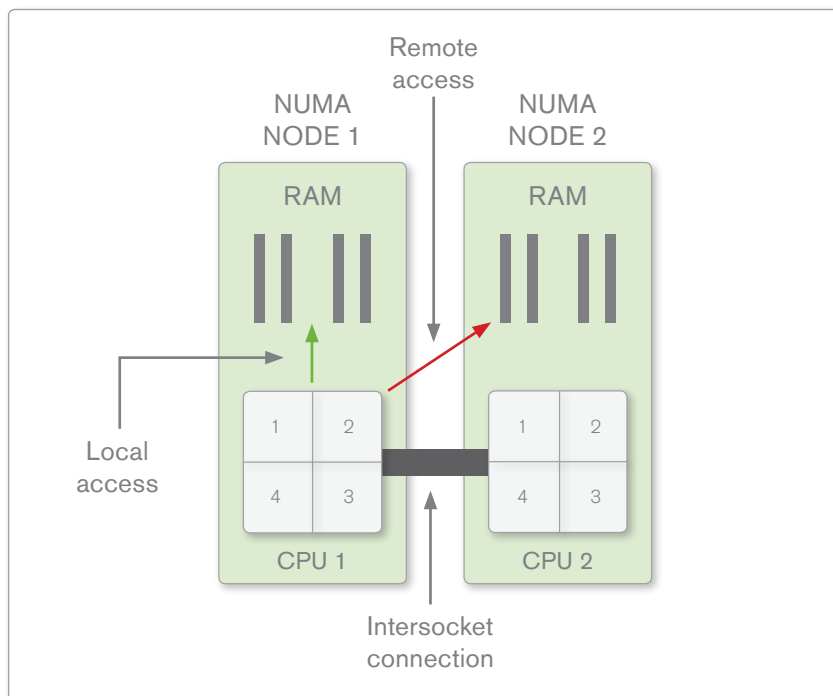
Figure 1 – Representation of NUMA Nodes

Test Environment

The Scalability Center has conducted performance tests focusing solely on NUMA and its impact on various environments, applications, and usage patterns. A combination of four tests were performed with NUMA disabled or enabled from within the hardware's BIOS settings and through QlikView's soft-switch setting.

Two physical servers (Table 1) were used to compare results from the impact of the NUMA setting with regards to the size and architecture of the server. QlikView 11 was installed on the servers for this test. A separate machine with the load client utilizing JMeter was used to simulate and initiate QlikView requests to these two servers.

Machine Name	SC-hptemp	SC-lab02
Server	HP ProLiant DL580 G7	HP ProLiant DL380 G7
CPU	32 Cores (4x8) @2.27GHz	12 Cores (2x6) @3.33GHz
RAM	512 GB RAM	144 GB RAM

Table 1 – Hardware used for Scalability Center tests

Testing Scenarios

The performance tests consisted of simulating virtual users using the Scalability Center QVScriptGenTool following three different scenarios:

- **HiTLoC:** Simulates high load with a low amount of cache hits. 50 concurrent users performing randomized selections constantly (no think-time between selections) using a 400mb application. This test stresses the system due amount of requests received.

- **MO:** This scenario uses the same application as the first scenario (**HitLoC**) but with 10 users performing selections that require heavy calculations by the server.

- **2.3:** This test demonstrates the impact of a small/medium QlikView application with 50 million rows. It uses a sophisticated dashboard with 7 tabs, dozens of charts and detail tables for granular level data analysis. This test simulates a more reasonable usage pattern compared to **HiTLoC** and **MO** which are designed to test specific areas that are performance sensitive.

Test Results

The following tables and charts present performance metrics for throughput (actions/minute), average response times, and CPU utilization as a result of the aforementioned tests.

Each metric is presented with a chart and a detail table for comparison. The results have been divided and represent the following:

- NUMA1_SS1 = NUMA & soft switch both ON
- NUMA1_SS0 = NUMA ON & soft switch OFF
- NUMA0_SS0 = NUMA & soft switch both OFF
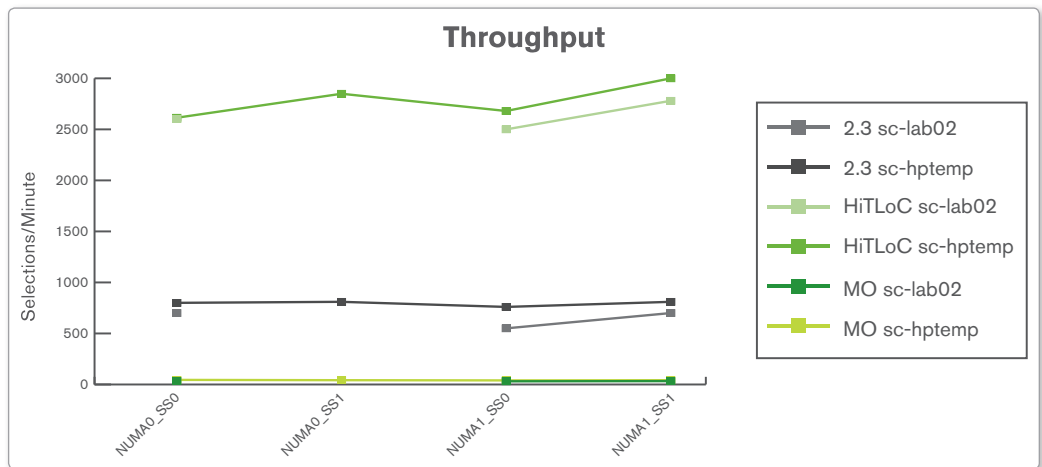- NUMA0_SS1 = NUMA OFF & soft switch ON

Figure 2 – Throughput – Selections / Minute

	2.3 sc-lab02	2.3 sc-hptemp	HiTLoC sc-lab02	HiTLoC sc-hptemp	MO sc-lab02	MO sc-hptemp
NUMA0_SS0	700	800	2600	2614	34	45
NUMA0_SS1	n/a	810	n/a	2849	n/a	42
NUMA1_SS0	550	760	2500	2680	31	40
NUMA1_SS1	700	810	2780	3000	33	42

Table 2 – Throughput – Selections / Minute

Higher values for throughput are better. The results are very similar when compared to the various settings. Lower values for throughput represent NUMA on and no soft-switch, which is expected. For the other setting combinations, they differ somewhat between types of tests. However, because of similar results, it can be said that the soft-switch is effective even if NUMA has been turned on. Results can be compared equally to those tests where NUMA was turned off or in some regards even better.


Response Times
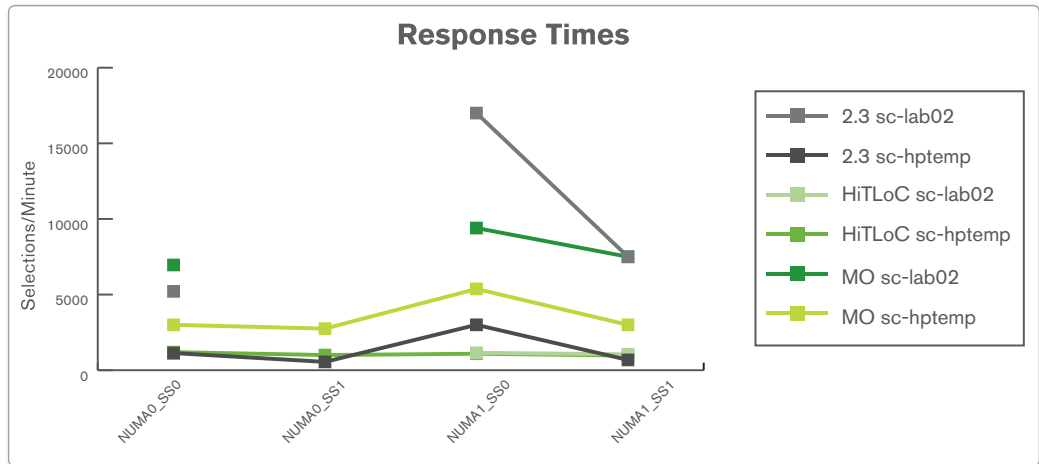
	2.3 sc-lab02	2.3 sc-hptemp	HiTLoC sc-lab02	HiTLoC sc-hptemp	MO sc-lab02	MO sc-hptemp
NUMA0_SS0	5600	1133	1100	1196	6700	3000
NUMA0_SS1	n/a	550	n/a	1002	n/a	2750
NUMA1_SS0	17000	3000	1150	1090	9400	5375
NUMA1_SS1	7500	680	1050	970	7500	3000

The average response times from the conducted tests show a bit higher variance when compared to the throughput test. The reason for some of the tests having higher response times in proportion to throughput is the think-times between actions which is obfuscating the real response time of a selection.

It stands very clear that NUMA enabled systems (set to on) impact the response times greatly, where at the same time, enabling the soft-switch will yield results similar to that of NUMA turned off.

Note that on the larger machine, sc-hptemp, enabling the soft-switch seems to have better performance regardless of the NUMA setting in the BIOS.

CPU Utilization
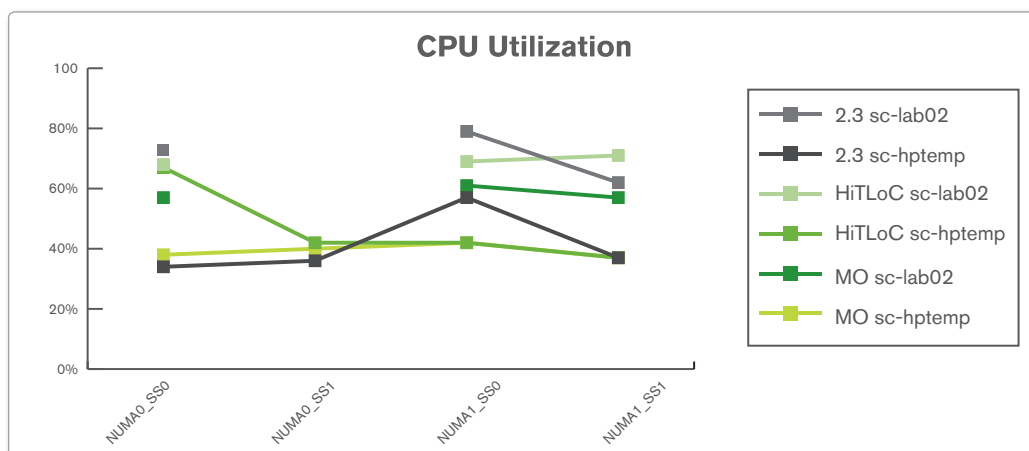
	2.3 sc-lab02	2.3 sc-hptemp	HiTLoC sc-lab02	HiTLoC sc-hptemp	MO sc-lab02	MO sc-hptemp
NUMA0_SS0	73%	34%	68%	67%	57%	38%
NUMA0_SS1	n/a	36%	n/a	42%	n/a	40%
NUMA1_SS0	79%	57%	69%	42%	61%	42%
NUMA1_SS1	62%	37%	71%	37%	57%	37%

Regarding CPU, lowest utilization was, in general, achieved by enabling the soft-switch. In contrast, the highest CPU usage was seen when NUMA was on and soft-switch set to off. Each test, and with its different settings, did not yield that great of a difference between the results.

Conclusions

With this set of fairly demanding tests, we conclude that there are differences between the settings configured using the soft-switch in the settings.ini and/or NUMA enabled BIOS. The worst possible performance is received by enabling NUMA and disabling soft-switch. Please take note that if NUMA is enabled on the server, it can be ignored by QlikView by enabling the soft-switch.