# Speeding Applications in Data Center Networks

## The Interaction of Buffer Size and TCP Protocol Handling and its Impact on Data-Mining and Large Enterprise IT Traffic Flows

DR160210F
February 2016

Miercom
www.miercom.com

# Contents

# 1 – Executive Summary

"How can I speed up my network data?" The answer is not a simple one, as there are at least a dozen parameters that collectively impact how fast a data file can move from one point to another. Some of these you can't do much about. Propagation delay, for example – the speed-of-light delay. It will always take longer to move the same file across country than between servers within a data center, especially with a connection-oriented protocol like TCP. But don't look to change TCP either: The dynamic duo of TCP and IP was standardized decades ago for universal Internet interoperability and can't readily be altered either.

But there are aspects of data networks that the user can change or modify to improve performance, such as in the architecture and features of the switches and routers selected. Switch-routers are not all created equal: The optimum architecture and feature set for traffic handling within a data center are quite different from a LAN-WAN router – or a remote, branch-office switch.

One hotly debated question is about one aspect of the data center network design. How big a buffer should be, or what kind of buffering functions needed on the switches in order to best support applications that involves bursty traffic. Would big buffer help or make it worse for incast /microburst traffic?

Miercom was engaged by Cisco Systems to conduct independent testing of two vendors' top-of-the-line, data-center switch-routers, including the Cisco Nexus 92160YC-X and Nexus 9272Q switches and the Arista 7280SE-72 switch. These switch products represent very different buffer architectures in terms of the buffer sizes and the buffer management. The objective: given the same network-intensive, data-center application and topology, the same data flows and standard network and transport protocol, how do they compare in terms of how fast TCP data flows are completed?

With the real-world traffic profiles, our testing results show that the Cisco Nexus 92160YC-X and Nexus 9272Q switches outperformed the Arista 7280SE-72 switch for congestion handling. The Arista 7280SE-72 switch has much deeper buffer than the Cisco switches, but with the built-in intelligent buffer management capabilities, both Cisco switches demonstrated clear advantage in flow completion time for small/medium flows over the Arista 7280SE-72 switch and provided the same or similar performance for large-size flows, which resulted in overall higher application performance.

The environment tested here incorporated and focused on:

- A specific environment: A high-speed data-center network linking 25 servers.
- Two specific traffic profiles from production data centers:
  - A data mining application
  - A typical large enterprise IT data center
- Data flows of widely varying sizes, from under 100 KB to over 10 MB, heavy buffer usage, and link loads to 95 percent.
- Standard New Reno TCP flow control as the transport layer protocol
- The key metric: How long to complete data transfers (flow completion times)?

What was tested, and how, and results relating to switch-router buffer size and TCP data-flow management are detailed in this paper.

Robert Smithers
CEO
Miercom

# 2 - TCP Congestion Control versus System Buffer Management

This study focused on two components of networking, which together play a major role in how fast it takes a "flow" – a data transfer, like an email message, a file transfer, a response to a database query, or a web-page download – to complete. They are:

- **TCP congestion control**. The Transmission Control Protocol (TCP) is the Layer-4 control protocol (atop IP at Layer 3) that ensures a block of data that's sent is received intact. Invented 35 years ago, TCP handles how blocks of data are broken up, sequenced, sent, reconstructed and verified at the recipient's end. The congestion-control mechanism was added to TCP in 1988 to avoid network congestion meltdown. It makes sure data transfers are accelerated or slowed down, exploiting the bandwidth that's available, depending on network conditions.
- **System buffer management**. Every network device that transports data has buffers, usually statically allocated on a per-port basis or dynamically shared by multiple ports, so that periodic data bursts can be accommodated without having to drop packets. Network systems such as switch-routers are architected differently, however, and can vary significantly in the size of their buffers and how they manage different traffic flows.

## TCP Congestion Control

To appreciate the role TCP plays, it's helpful to understand how congestion control is managed. TCP creates a logical connection between source and destination endpoints. The actual routing of TCP data, finding how to make the connection and address packets, is relegated to the underlying IP protocol layer.

Congestion is continually monitored on each separate connection that TCP is maintaining. A built-in feedback mechanism lets TCP determine whether to send more packets, and use more network bandwidth, or to back off and send less packets due to congestion. A destination acknowledges packets received by sending back "ACK" messages, indicating receipt. With each ACK, TCP can incrementally increase the pace of sending packets to use any extra available bandwidth.

Similarly, TCP deduces that a packet has been lost or dropped after receiving three duplicate ACKs – without an acknowledgement for a particular outstanding packet. TCP then considers the packet loss as a congestion indication, and then backs off and cuts its transmission rate, typically in half, to reduce the congestion in the network.

## Buffering Issues with TCP

The inherent congestion control of TCP, and the buffering in a system that TCP data traverses enroute, interact with each other to form a feedback loop. However, the two mechanisms serve different purposes: The TCP congestion control aims to continuously probe network capacity using packet drops as indications to slow down, while network buffering is to avoid packet drops as much as possible.  In fact, the two don't necessarily always complement each other, or expedite data, and may actually work at cross purposes.

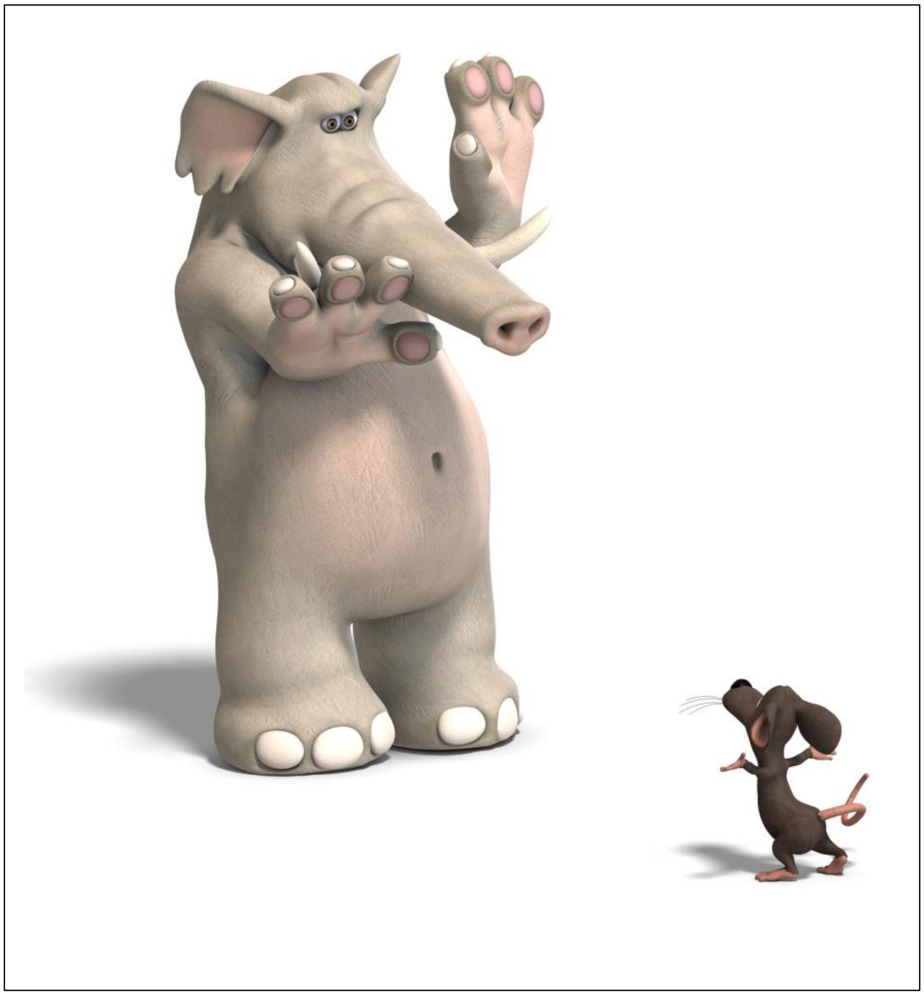
TCP doesn't know when its data is being buffered by a system on its way to a destination.  In a congested situation, TCP data may back up in a buffer queue.  In that case not only is the TCP data delayed, but so is any congestion feedback that TCP could use for congestion control.

Buffering, is advantageous to speedy data-delivery performance when traffic flows are small and bursty.  In this case, because data moves on quickly and any added delay due to buffering is very brief, TCP does not receive enough missing-packet indications to have to react to congestion.  An example of this traffic type is "incast," where many stations transmit to one simultaneously, such as in response to a broadcast-type big-data query.  All of these would be considered "mice" flows.

As a rule, TCP congestion control works efficiently when buffering is adequate to drive full link utilization and the added latency (delay) imposed by the buffering is small. Any excessive buffering is not advantageous, for large, sustained long-term flows.  Network engineers term these traffic types as "elephant flows," in comparison with "mice flows."  Excessive buffering for such large flows does not improve throughput.  What's more, it delays the congestion feedback that TCP relies on to regulate its transmission rate upon network congestion.

# 3 - TCP Flow Sizes: Mice and Elephants

In production data-center networks, traffic is inevitably a mixture of big and small flows. A larger percentage of the flows, studies show, are typically small flows, though most of the traffic load (percentage of total volume) is contributed by the big flows.



In both traffic profiles we tested, the data-mining applications and the large enterprise IT, there are both big flows (elephants), which send large multi-packet data across TCP connections, and small flows (mice). While elephants account for fewer flows, each is responsible for large amounts of data.

The mice are smaller, bursty flows, such as where one query is sent to many servers and results in lots of small messages being sent back, over many TCP connections, from the remote servers, back to the single host that originated the query. These small messages may require only three to five TCP packets. In such a case TCP's congestion control – backing off and retrying messages – may not even be invoked since it takes three duplicate ACK messages for TCP to conclude a packet is missing and needs to be retransmitted. Alternatively, a retransmission would also occur on a retransmission timeout, which takes much longer – on the order of 200 milliseconds (ms). The latter case is detrimental. A good solution to avoid it is to prioritize small flows, to avoid the longer delays in detecting packet drops and network congestion.

For the elephant flows, it is better to let TCP handle the congestion control in its normal fashion. TCP will adjust the rate of transmission based on its receipt of packet ACKs. It may speed up the flow if the ACKs come back quickly, or slow the flow if packet drops are detected. Excessive buffering the data in a large flow just obscures the actual traffic characteristics from TCP – and consumes buffer space that could be better used servicing the small mice flows.

## Deep Buffer versus Intelligent Buffer

Buffering is used to absorb instantaneous traffic bursts, to avoid packet drops during network congestion. A common practice is to put in as much buffer as possible. However, since the buffer space is a common resource shared by the inevitable mixture of elephant and mice flows, how to use this shared resource can significantly impact applications' performance.

Users may face two architectural alternatives in the switch-router systems they procure:

- Deep buffer or
- Intelligent buffer

With a simple buffer, whether relatively small or "deep" (large), data is placed in the buffer on a first-come, first-served basis, and then moved out in the same order as port transmission capacity comes available.

The currently available deep buffer architecture does not distinguish flows based on their sizes. All flows are subject to the same queuing and scheduling policies. In other words, elephant flows and mice flows are treated the same. However, due to the nature of TCP congestion control, elephant flows continuously increase their sending rate and cause network buffers to fill up until TCP receives a packet drop or mark when the buffer is full or nearly full. As a result, these flows will consume most of the buffer space and starve mice flows even with a deep buffer.

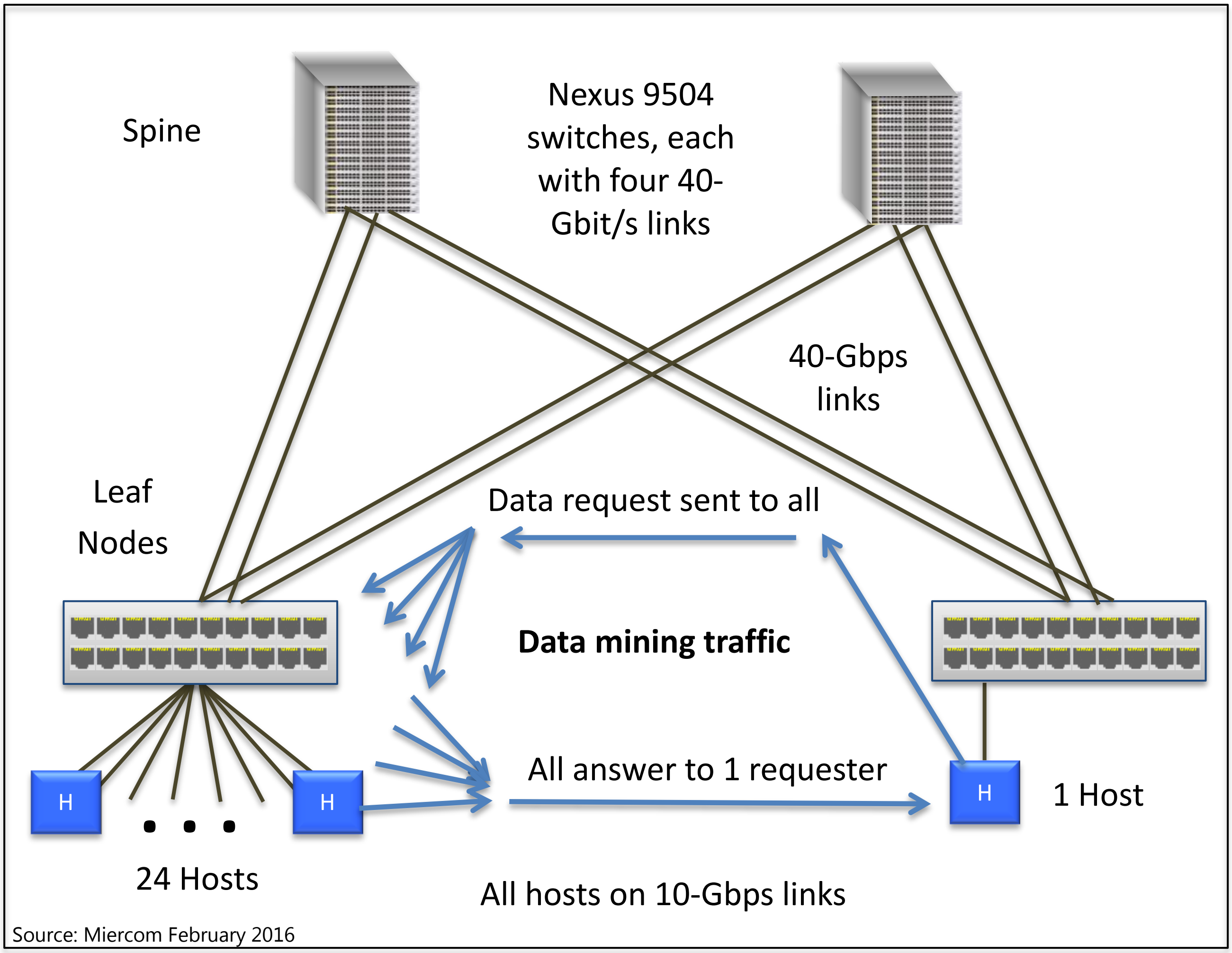
In addition, even if a small flow is lucky and is put into a queue (rather than be dropped), it will experience much longer added latency while waiting in queue due to a largely occupied buffer. The deeper the buffer, the longer the queue and the longer the latency. So more buffer does not necessarily guarantee better small-flow performance, it often leads to longer queuing delay and hence longer flow completion time.

Therefore, no one benefits from simple deep buffering: mice flows aren't guaranteed buffer resources and can suffer from long queuing delays and bandwidth hungry elephant flows suffer because large buffers do not create more link bandwidth.

In contrast, intelligent buffer architecture manages buffer and queue scheduling intelligently, identifying and treating large and small flows differently. Mice flows are expedited and not dropped to avoid detrimental timeouts and elephant flows are given early congestion notifications through TCP to prevent them from occupying unnecessary buffer space. As a result, intelligent buffering allows elephant and mice flows to share network buffers more gracefully. There is buffer space for the mice flow bursts and elephant flows are regulated to fully utilize the link capacity.

## 4 - Test Bed: Data Mining and Buffers

To see how different switch-routers perform in terms of flow completion times – the objective of this study – a realistic test bed, representing a data-center network infrastructure, was assembled see diagram below.


Spine
Nexus 9504 switches, each with four 40-Gbit/s links
40-Gbps links
Leaf Nodes
Data request sent to all
Data mining traffic
All answer to 1 requester
H
H
H
1 Host
24 Hosts
All hosts on 10-Gbps links

Source: Miercom February 2016

The test-bed data-center network featured two “spine” switches – Cisco Nexus 9504 modular switches. Each supported four 40-Gbps links to two “leaf” nodes, as indicated.

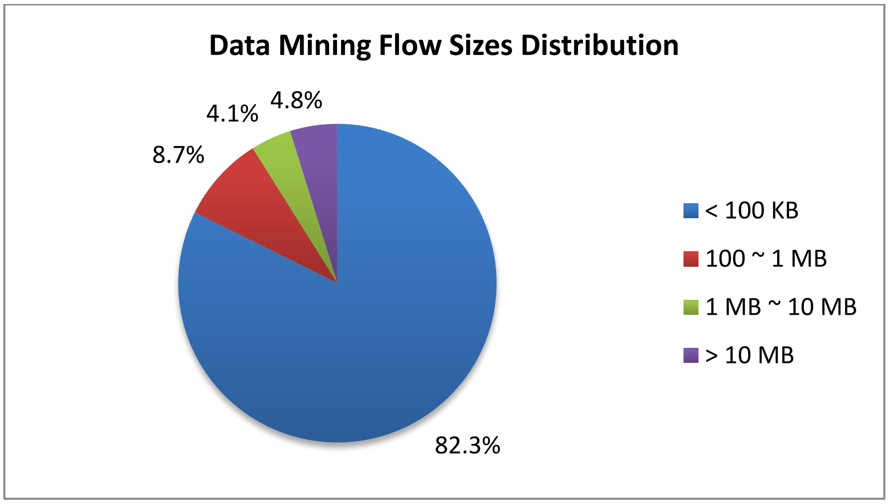
The “leaf” nodes were the devices tested. Three switches were tested, one pair at a time:

- Cisco Nexus 92160YC-X
- Cisco Nexus 9272Q
- Arista 7280SE-72.

Cisco UCS C200 High-Density Rack- Mount Servers were used as application hosts.  One leaf switch connected to 24 server hosts, each via a 10-Gbit/s link.  A single 25th server host connected to the other leaf switch, also via a 10-Gbit/s link.  This server was the data-mining host, sending requests to remote server hosts and then receiving answers back from all 24 hosts. The 24 host servers exchanging traffic, local and remote stressed the switch's egress (outbound) buffer.

## About the Tests and Flows Applied

The traffic applied in the testing was modelled to reflect real-world data-mining applications. Elephant flows comprised just about 5 percent of the number of flows, but 95 percent of the total data volume.  The following chart shows the TCP flow size distribution for this data-mining application. Custom scripts were written to create random flows, which collectively followed distribution based on these percentages.



Multiple copies of these scripts were run to load the 10-Gbit/s link on the requestor host server to 20-percent load (2 Gbit/s).  This was the first test run.  Subsequent passes ran additional copies of the script to achieve 30-, 40-, 50-, 60-, 70-, 80-, 90- and 95-percent load on the requestor host's port.

The key metric sought from this testing was Flow Completion Time, or FCT – the time it took each flow to finish. Flow completion time is used instead of typical metrics such as: average link unitization, packet drops and queuing latency. As FCT directly reflects an application's ultimate performance and it encompasses all the metric parameters above. The average FCT was captured by the script and included in the script output.

The script returned data for an overall average FCT for each of five flow size ranges: under 100 KB; 100 KB; 100 KB to 1 MB; 1 MB to 10 MB, and over 10 MB.  The below table shows the script output, in this case for the Cisco Nexus 92160YC-X, and the under 100 KB flow size ranges.

| Flow Sizes < 100 KB | | |
|---|---|---|
| % Load & Flow size | Flow Completion Time (msec) | # flows |
| 20 | 0.24 | 5213 |
| 30 | 0.24 | 7819 |
| 40 | 0.24 | 10486 |
| 50 | 0.24 | 13092 |
| 60 | 0.24 | 15699 |
| 70 | 0.24 | 18364 |
| 80 | 0.25 | 20971 |
| 90 | 0.25 | 23577 |
| 95 | 0.30 | 24940 |

The same output was generated for the other four flow size ranges. Each test was run for 5 minutes. The output from a five-minute test was compared to a 10 minute test and the results were equivalent.

## Buffers and Buffer Utilization

The buffer utilization of the switches was also carefully monitored during the tests. The below table shows the buffers allotted per port, according to vendor documentation, and the maximum amount of buffer used during the testing.

| Leaf switch tested | **Cisco Nexus 9272Q** | **Cisco Nexus 92160YC-X** | **Arista 7280SE-72** |
|---|---|---|---|
| Documented buffer size | Max 5 MB / port | Max 10 MB / port | 50+ MB / port |
| Max buffer in use during testing | 2.97 MB | 2.95 MB | 52 MB |
| Intelligent Buffer capability | Yes | Yes | No |

The max buffer in use values were obtained via command-line queries while the tests were running. The largest observed values across all test runs are shown here.

The size difference between the Cisco buffers and the Arista buffer is considerable. The Arista buffer size in the Arista switch in use in this test is at least 10 times larger than the Cisco Nexus 9272Q buffer, and at least 5 times larger than the Cisco 92160YC-X's. Note, too, that the Cisco switches support Intelligent buffering – the ability to prioritize small flows over large flows.

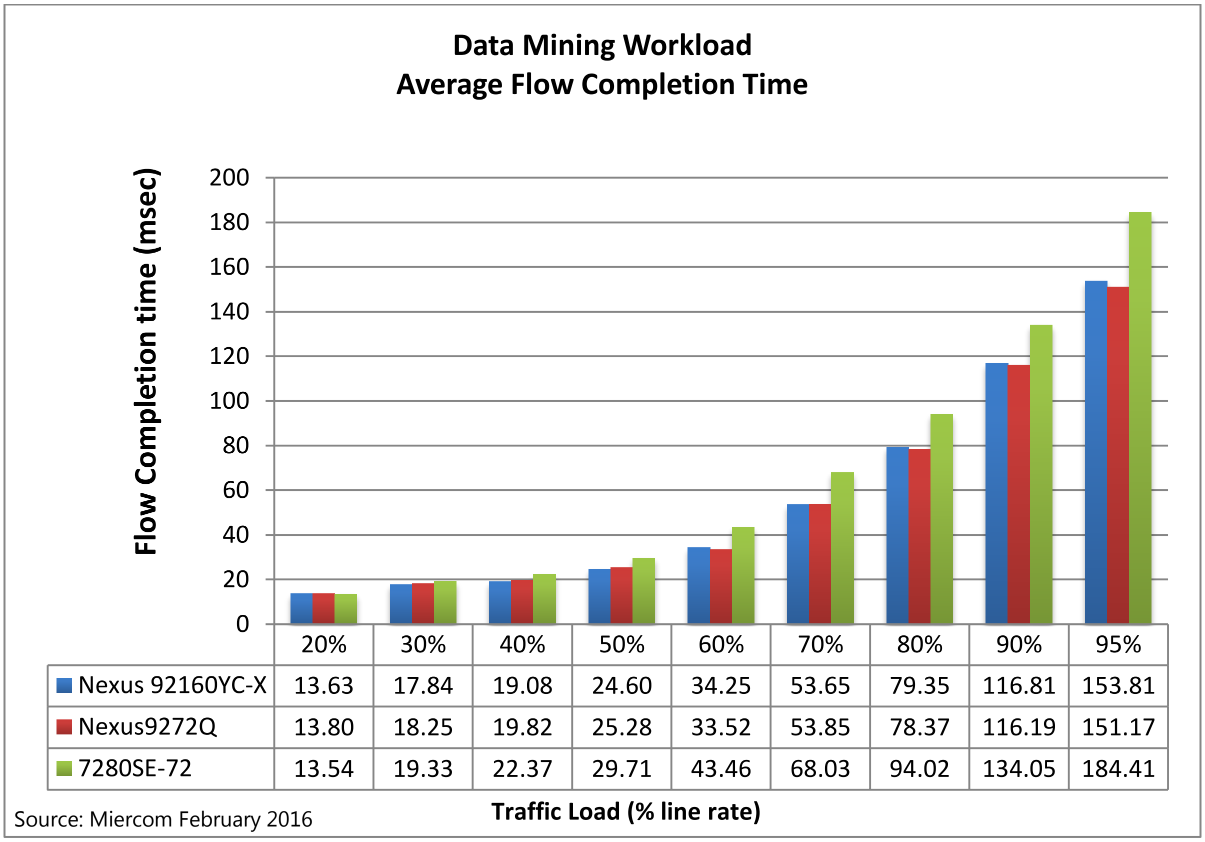
The testing was conducted with a Cisco-provided pre-release image for a preview of the intelligent buffer features. The Nexus 92160YC-X and Nexus 9272Q switch hardware have the functionalities and the software support of the feature will become available in a subsequent NX-OS release.

The default buffer configuration and settings were used in all cases, for all the switches tested.

# 5 – Results

## Average Flow Completion Times for All Flows

With all flow sizes averaged, the differences in Flow Completion Times are barely discernible below about 40 percent link load.



| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 13.63 | 17.84 | 19.08 | 24.60 | 34.25 | 53.65 | 79.35 | 116.81 | 153.81 |
| 🟥 Nexus9272Q | 13.80 | 18.25 | 19.82 | 25.28 | 33.52 | 53.85 | 78.37 | 116.19 | 151.17 |
| 🟩 7280SE-72 | 13.54 | 19.33 | 22.37 | 29.71 | 43.46 | 68.03 | 94.02 | 134.05 | 184.41 |

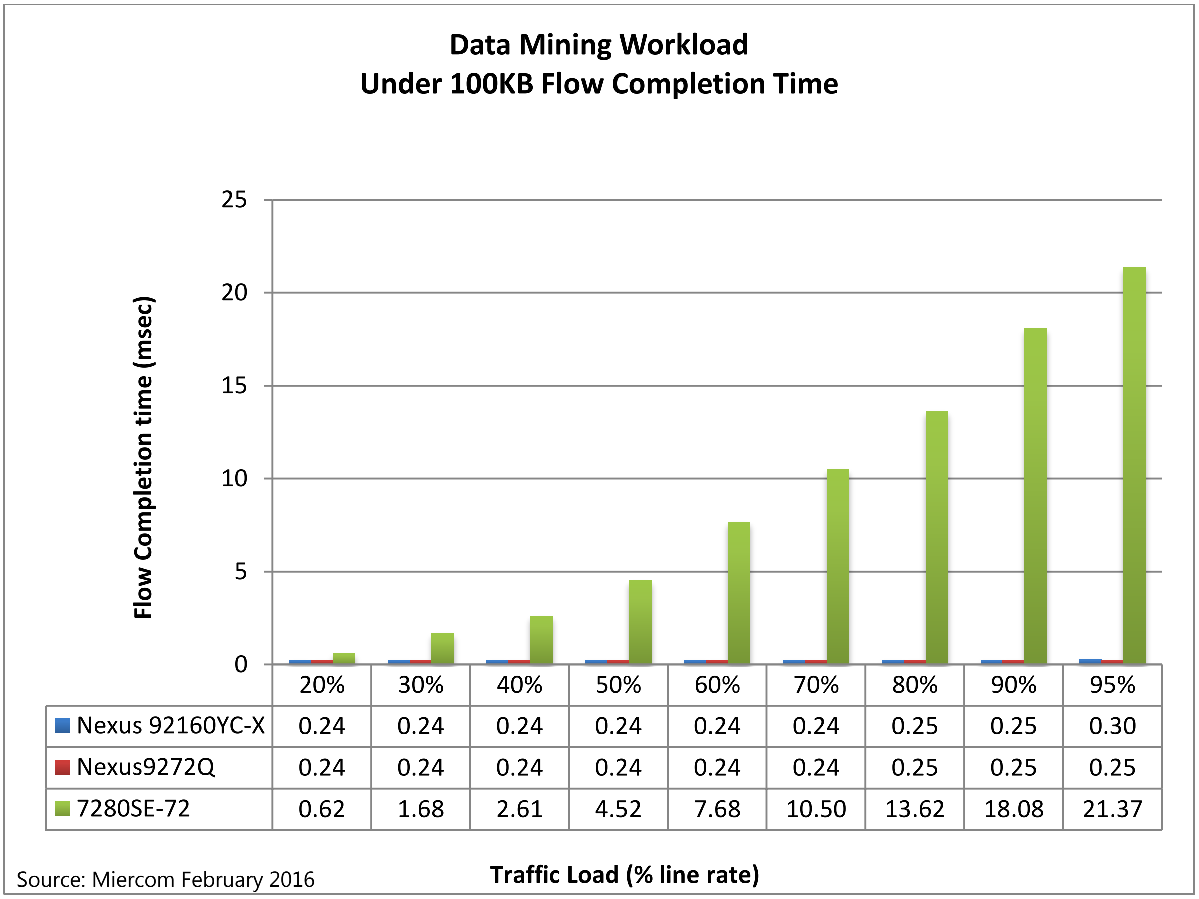
Source: Miercom February 2016

The average Flow Completion Times are essentially the same for both the Cisco Nexus 9272Q and Nexus 92160YC-X switches. Above 50 percent link load, however, as buffers become more heavily loaded, the longer Flow Completion Times for the Arista 7280SE-72 become more pronounced.

Beyond 60 percent load, Flow Completion Times for the Arista switch are about 20 to 30 percent longer, on average, than with either Cisco switch, even though the Arista 7280SE-72 switch uses ten times the buffer storage than the Cisco Nexus 9272Q switch.

## Mice: Under 100-KB Flow Completion Times

The most pronounced difference in Flow Completion Times between the Cisco and Arista switches occurs when we focus on the smallest flow sizes, under 100 Kilobytes, as shown in the following graph.



| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 0.24 | 0.24 | 0.24 | 0.24 | 0.24 | 0.24 | 0.25 | 0.25 | 0.30 |
| 🟥 Nexus9272Q | 0.24 | 0.24 | 0.24 | 0.24 | 0.24 | 0.24 | 0.25 | 0.25 | 0.25 |
| 🟩 7280SE-72 | 0.62 | 1.68 | 2.61 | 4.52 | 7.68 | 10.50 | 13.62 | 18.08 | 21.37 |

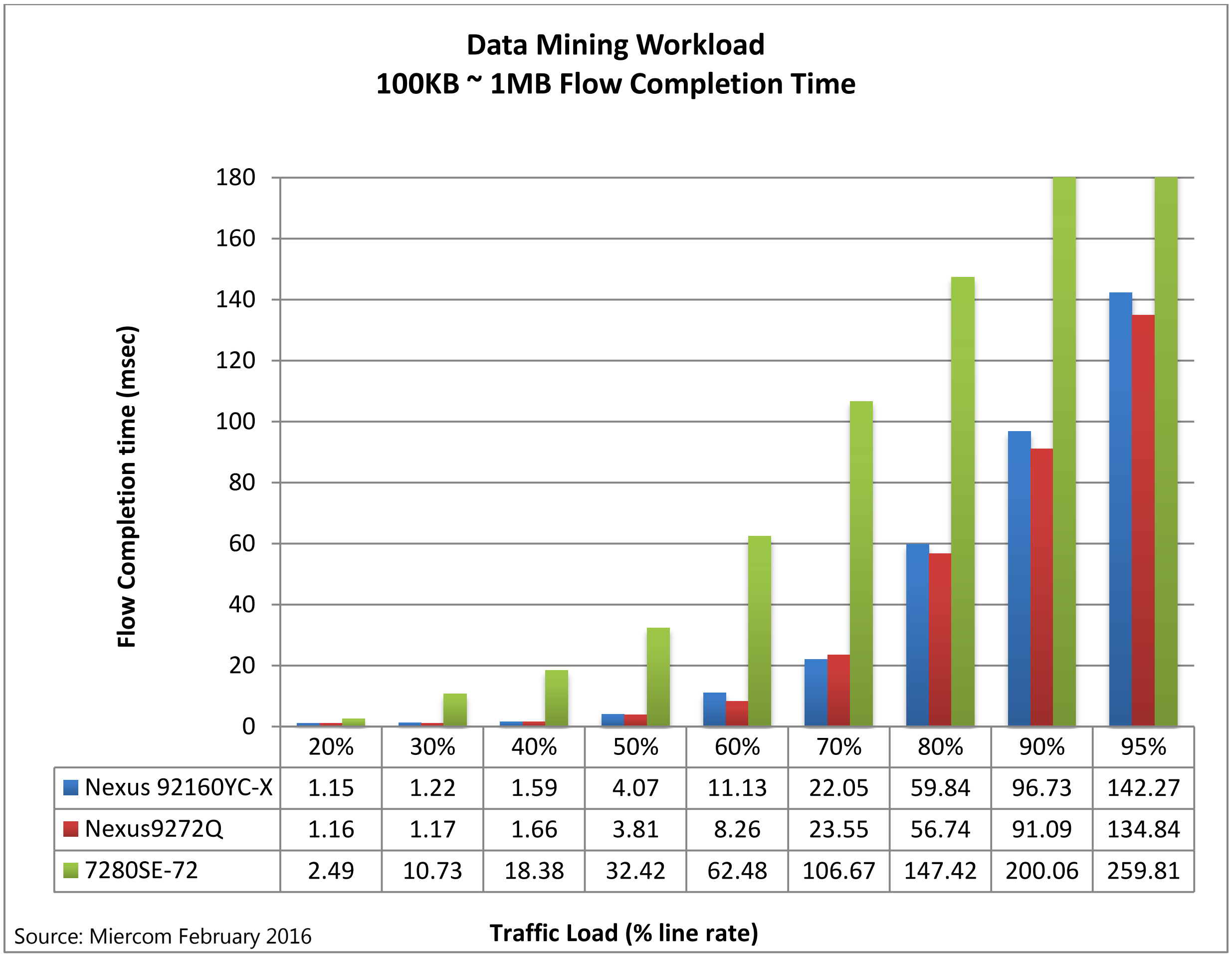
Source: Miercom February 2016

The small bursty flows complete much faster through the Cisco switches than the Arista, particularly as the traffic load increases. At 95% traffic load, it is two orders of magnitude better than Arsita. The much larger Arista buffer doesn't help, and the data suggests this actually is an impediment. Additionally, the Cisco switch's Intelligent Buffer algorithm ensures that mice get through ahead of the large flows.

Because of the packet prioritization for mice flows over elephant flows on the Cisco switches, the mice flows' completion time remains extremely low, around 0.25 msec, and did not change as traffic load increases. At traffic loads above about 40 percent, Flow Completion Times became one order of magnitude longer with Arista than the Cisco 92160YC-X switch, and at the 95% load, the flow completion time with the Arista 7280SE-72 switch became closer to two order of magnitude longer than that with the Cisco Nexus 92160YC-X or Nexus 9272Q switches. The data indicates that the larger the buffers, the longer small-sized flows take to complete.

## 100-KB to 1-MB Flow Completion Times

For small-to-medium-sized flows – in the range from 100 Kilobytes to 1 Megabyte – the difference in Flow Completion Times between the Cisco and Arista switches are significant. At 30- to 70-percent traffic load, flows this size take 4 to 10 times longer to complete through an Arista switch than either Cisco switch. See graph below.



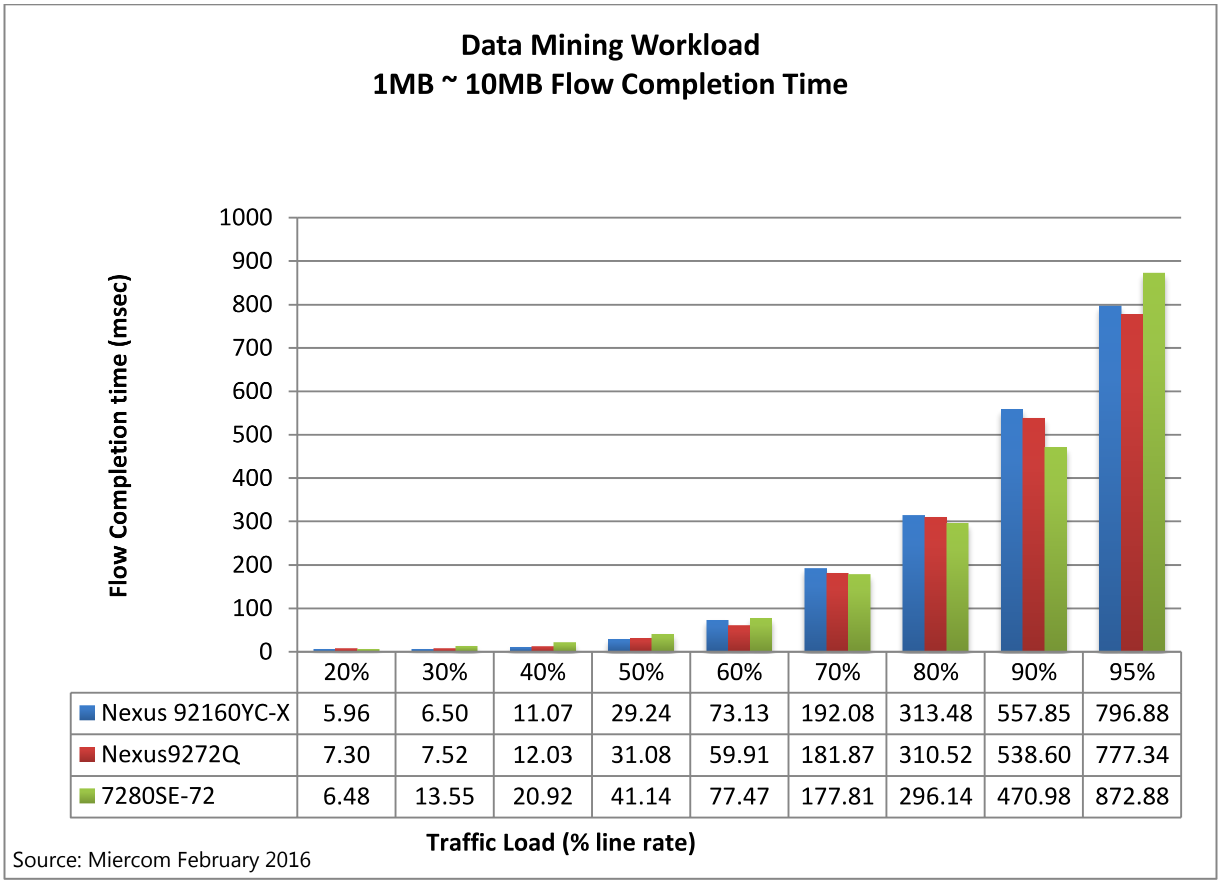
| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 1.15 | 1.22 | 1.59 | 4.07 | 11.13 | 22.05 | 59.84 | 96.73 | 142.27 |
| 🟥 Nexus9272Q | 1.16 | 1.17 | 1.66 | 3.81 | 8.26 | 23.55 | 56.74 | 91.09 | 134.84 |
| 🟩 7280SE-72 | 2.49 | 10.73 | 18.38 | 32.42 | 62.48 | 106.67 | 147.42 | 200.06 | 259.81 |

Source: Miercom February 2016

At 80 percent and higher traffic loads, flows through the Arista switch take at least twice as long to complete. The data indicates that connection-oriented TCP traffic speeds small-to-medium flows better than large buffers, which actually impedes Flow Completion Times.

## 1-to-10-MB Flow Completion Times

The difference in Flow Completion Times for mid-to-large-sized flows – 1- to 10-Megabytes – narrows between the Cisco switches and the Arista switch. This also demonstrates that the larger buffer sizes per port offer no added advantage to speed the completion times of the larger flows



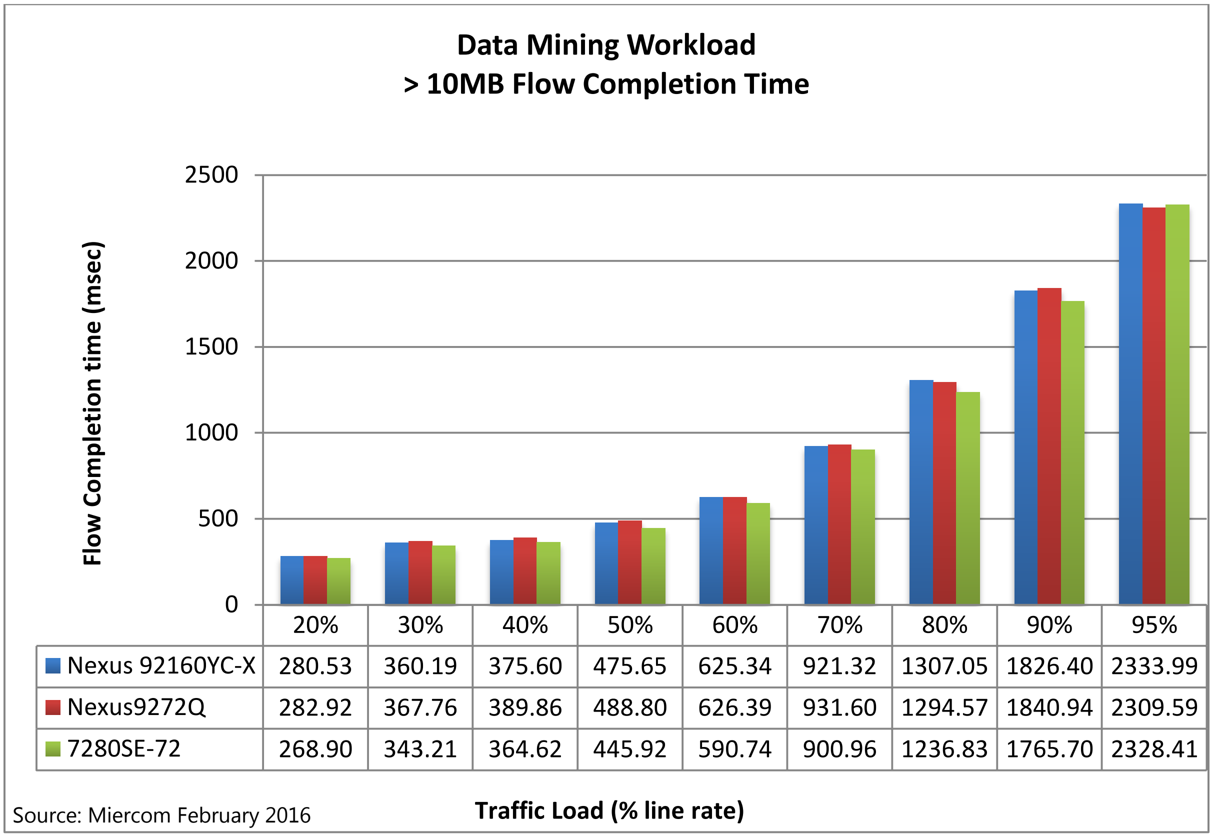
**Data Mining Workload**
**1MB ~ 10MB Flow Completion Time**

| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 5.96 | 6.50 | 11.07 | 29.24 | 73.13 | 192.08 | 313.48 | 557.85 | 796.88 |
| 🟥 Nexus9272Q | 7.30 | 7.52 | 12.03 | 31.08 | 59.91 | 181.87 | 310.52 | 538.60 | 777.34 |
| 🟩 7280SE-72 | 6.48 | 13.55 | 20.92 | 41.14 | 77.47 | 177.81 | 296.14 | 470.98 | 872.88 |

Source: Miercom February 2016

## Elephants: Over-10-MB Flow Completion Times

For Elephant flows, the smaller-buffered Cisco switches show essentially the same Flow Completion Times as the much larger-buffered Arista switch, with just a few percent difference, as shown in the graph below. Indeed at the largest flow sizes the larger Arista buffer size offers no advantage over the small, smart Cisco buffer.

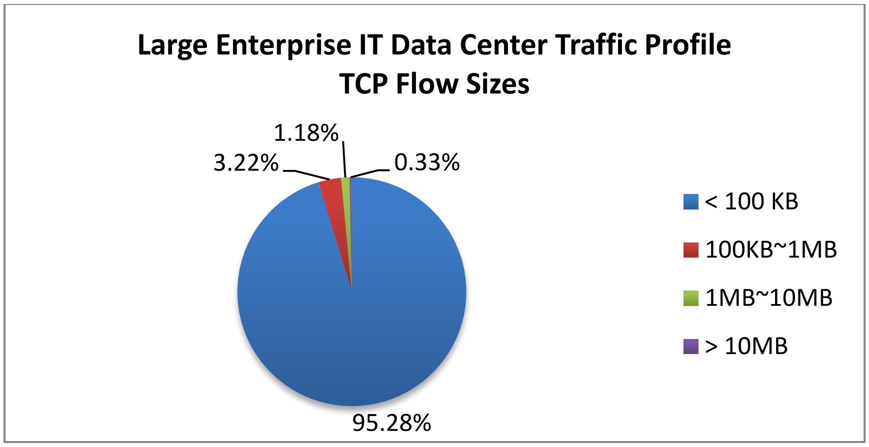


| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 280.53 | 360.19 | 375.60 | 475.65 | 625.34 | 921.32 | 1307.05 | 1826.40 | 2333.99 |
| 🟥 Nexus9272Q | 282.92 | 367.76 | 389.86 | 488.80 | 626.39 | 931.60 | 1294.57 | 1840.94 | 2309.59 |
| 🟩 7280SE-72 | 268.90 | 343.21 | 364.62 | 445.92 | 590.74 | 900.96 | 1236.83 | 1765.70 | 2328.41 |

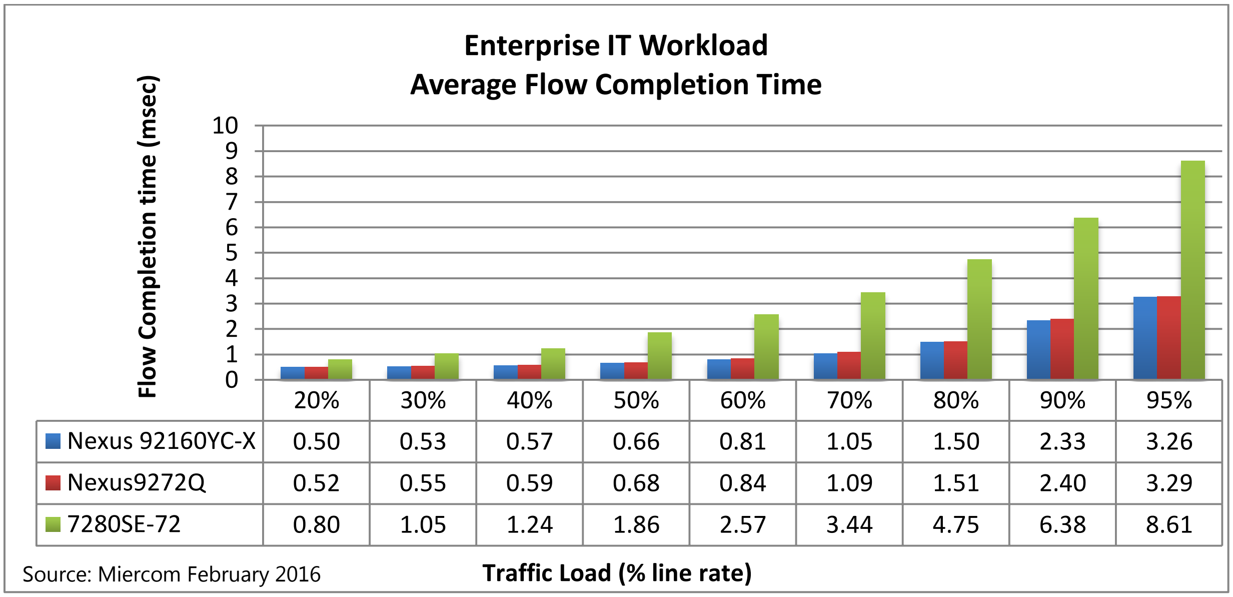
Source: Miercom February 2016

# 6 – Results with the Large Enterprise IT Traffic Profile

In addition to the data mining traffic profile, we also ran the tests with a typical large enterprise IT data center traffic profile that we gathered from analysis of a real data center. The following graph shows the flow sizes distribution.



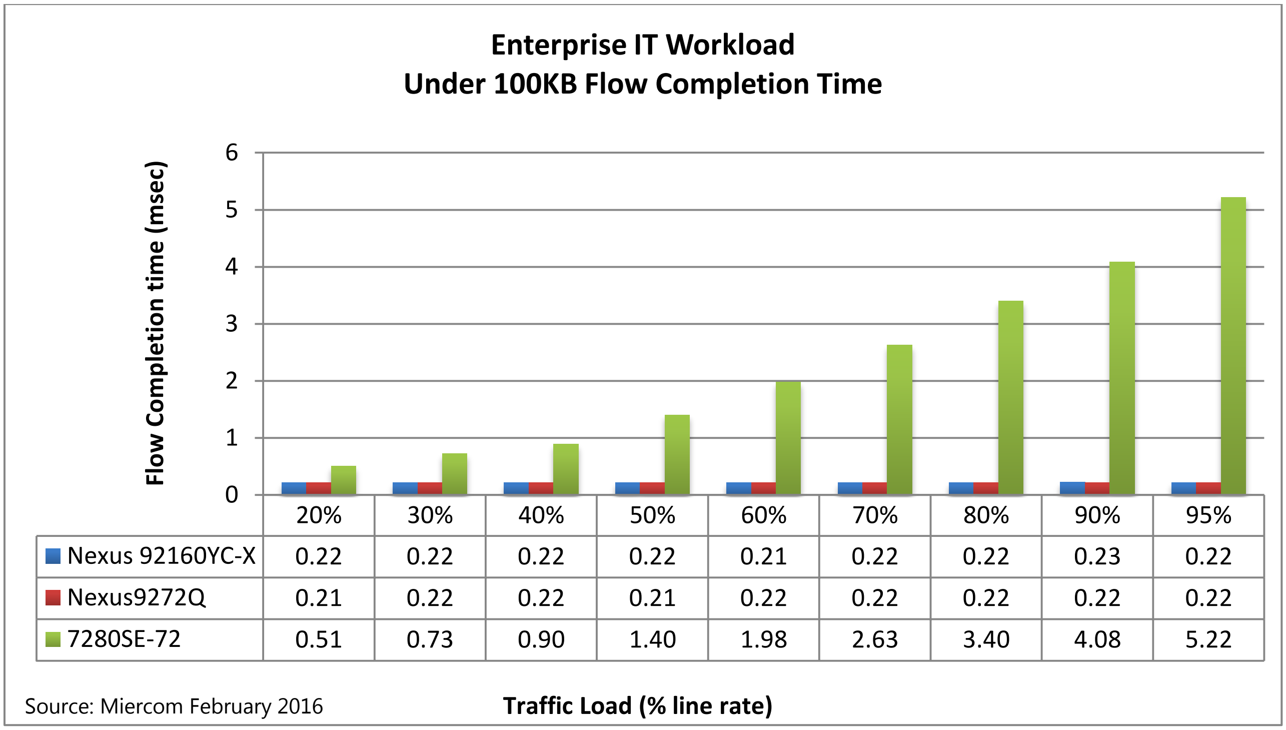
Below are the captured results, which show the similar trend of TCP performance benefits with the intelligent buffer on Cisco switches versus the deep buffer on the Arista switch. The memory utilization is similar to the one obtained in the previous test.



| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 0.50 | 0.53 | 0.57 | 0.66 | 0.81 | 1.05 | 1.50 | 2.33 | 3.26 |
| 🟥 Nexus9272Q | 0.52 | 0.55 | 0.59 | 0.68 | 0.84 | 1.09 | 1.51 | 2.40 | 3.29 |
| 🟩 7280SE-72 | 0.80 | 1.05 | 1.24 | 1.86 | 2.57 | 3.44 | 4.75 | 6.38 | 8.61 |

Source: Miercom February 2016

*With the flow sizes and load distribution in the typical large enterprise IT data center traffic profile, the overall average flow completion time on the Cisco switches were 30%~60% better than that on the Arista switch, varying with the traffic load. The heavier the traffic load, the more advantageous the Cisco's results were in comparison to those of the Arista switch.*

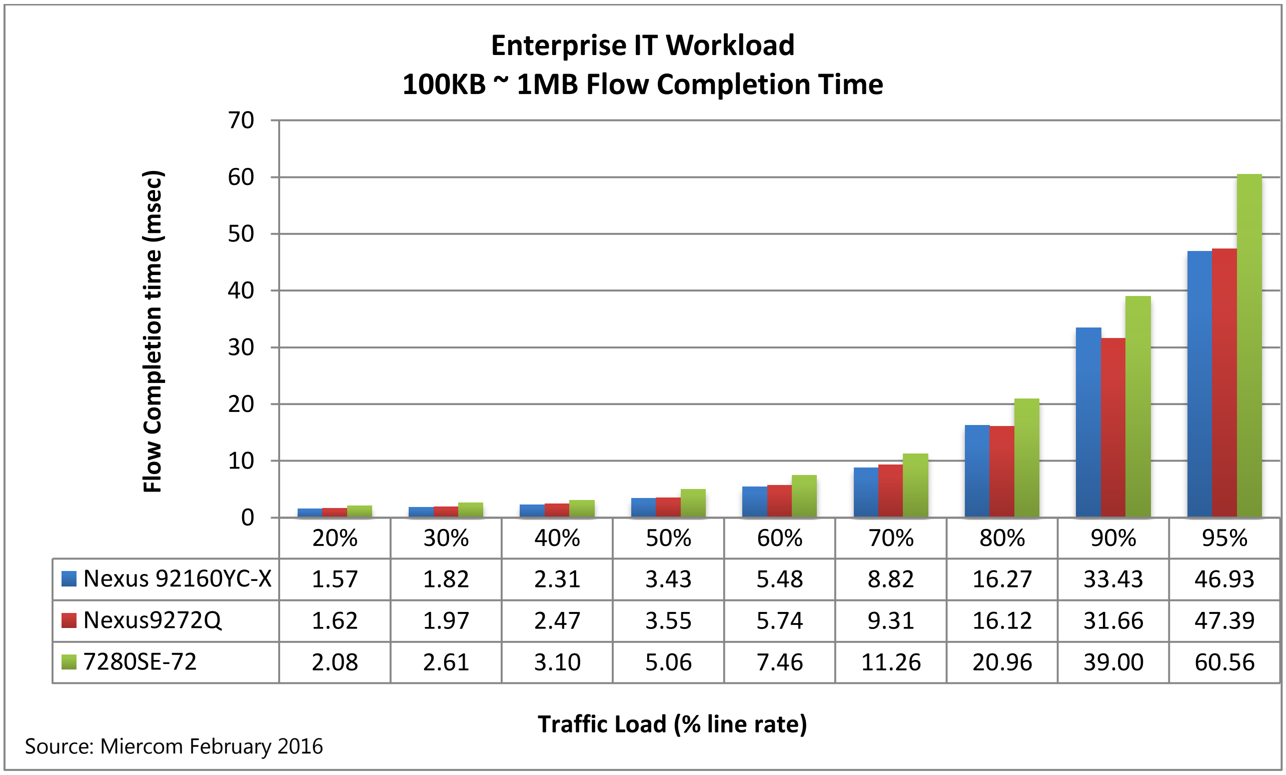


Enterprise IT Workload
Under 100KB Flow Completion Time

| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 0.22 | 0.22 | 0.22 | 0.22 | 0.21 | 0.22 | 0.22 | 0.23 | 0.22 |
| 🟥 Nexus9272Q | 0.21 | 0.22 | 0.22 | 0.21 | 0.22 | 0.22 | 0.22 | 0.22 | 0.22 |
| 🟩 7280SE-72 | 0.51 | 0.73 | 0.90 | 1.40 | 1.98 | 2.63 | 3.40 | 4.08 | 5.22 |

Source: Miercom February 2016

*With the small flow sizes (< 100 KB), the Cisco switches show absolute superior performance over the Arista switch. It was 57% better at 20% traffic load. As the traffic load increased to 70% and beyond, Cisco switches performed more than an order of magnitude better than Arista switches.*

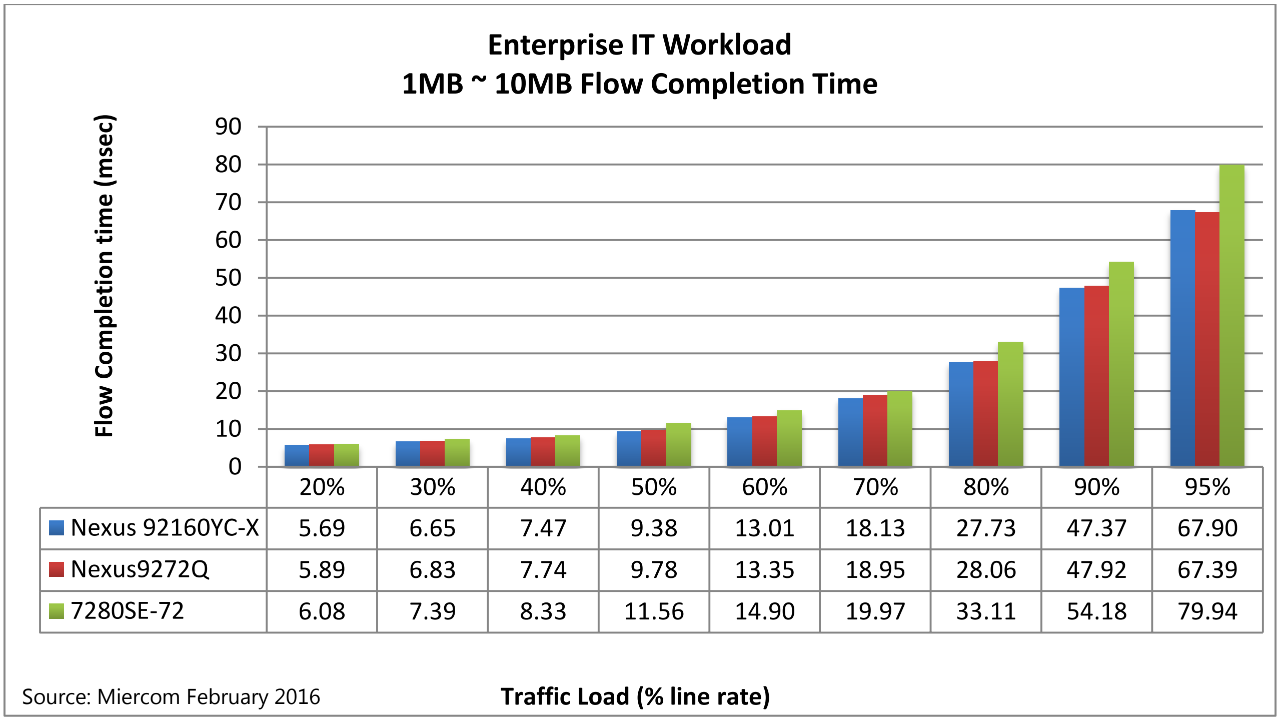


Enterprise IT Workload
100KB ~ 1MB Flow Completion Time

| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 1.57 | 1.82 | 2.31 | 3.43 | 5.48 | 8.82 | 16.27 | 33.43 | 46.93 |
| 🟥 Nexus9272Q | 1.62 | 1.97 | 2.47 | 3.55 | 5.74 | 9.31 | 16.12 | 31.66 | 47.39 |
| 🟩 7280SE-72 | 2.08 | 2.61 | 3.10 | 5.06 | 7.46 | 11.26 | 20.96 | 39.00 | 60.56 |

Source: Miercom February 2016

*With the small to medium flow sizes (100 KB-1MB), the Cisco switches proved significant performance. It was 15% - 30% better than what can be achieved by the Arista switch.*

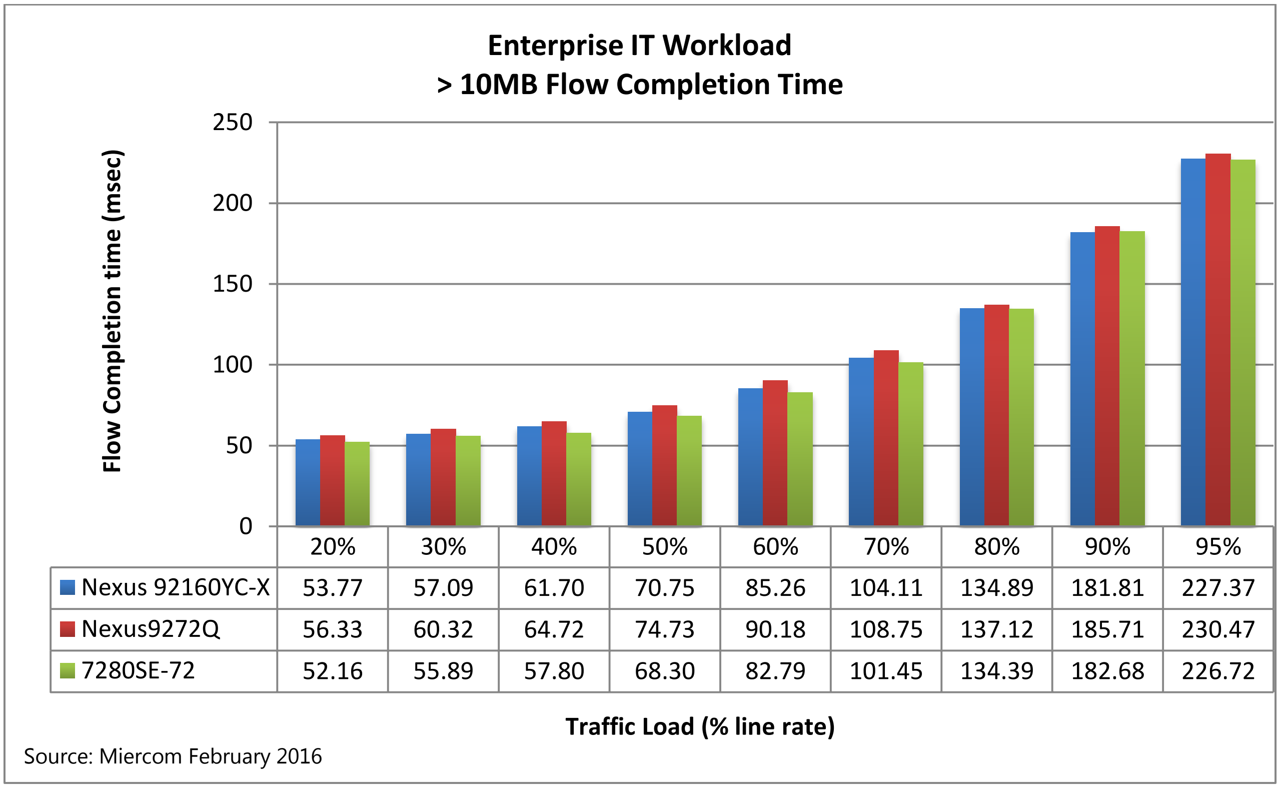


Enterprise IT Workload
1MB ~ 10MB Flow Completion Time

| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 5.69 | 6.65 | 7.47 | 9.38 | 13.01 | 18.13 | 27.73 | 47.37 | 67.90 |
| 🟥 Nexus9272Q | 5.89 | 6.83 | 7.74 | 9.78 | 13.35 | 18.95 | 28.06 | 47.92 | 67.39 |
| 🟩 7280SE-72 | 6.08 | 7.39 | 8.33 | 11.56 | 14.90 | 19.97 | 33.11 | 54.18 | 79.94 |

Source: Miercom February 2016

*The performance advantage by the Cisco switches extends to the medium-to-large size flows as well. For flows with sizes ranging from 1MB to10MB, the Cisco switches achieve better performance across various traffic load and the gains are still substantial: e.g. reaching 15% at the 95% traffic load.*



Enterprise IT Workload
> 10MB Flow Completion Time

| | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% | 95% |
|---|---|---|---|---|---|---|---|---|---|
| 🟦 Nexus 92160YC-X | 53.77 | 57.09 | 61.70 | 70.75 | 85.26 | 104.11 | 134.89 | 181.81 | 227.37 |
| 🟥 Nexus9272Q | 56.33 | 60.32 | 64.72 | 74.73 | 90.18 | 108.75 | 137.12 | 185.71 | 230.47 |
| 🟩 7280SE-72 | 52.16 | 55.89 | 57.80 | 68.30 | 82.79 | 101.45 | 134.39 | 182.68 | 226.72 |

Source: Miercom February 2016

*For the largest elephant flows with sizes larger than 10MB, Cisco switches achieve similar performance as the Arista switch, similar to the previous traffic profile.*

# 7 - Test Conclusions

Since mice flows are often mission critical (including, for example, control and alarm messages, Hadoop application communications, etc.), giving these flows a priority buffer pathway enables them to complete faster and their applications to perform better overall. The above test results show that expediting mice flows and regulating the elephant flows early under the intelligent buffer architecture on the Cisco Nexus 92160YC-X and 9272Q switches can bring orders of magnitude better performance for mission critical flows without causing elephant flows to slow down.

Intelligent buffering allows the elephant and mice flows to share network buffers gracefully: there is enough buffer space for the bursts of mice flows while the elephant flows are properly regulated to fully utilize the link capacity. Simple, deep buffering can lead to collateral damage in the form of longer queuing latency, and hence longer flow completion time for all flow types.

In addition, the results show that significant performance gains are achieved across multiple traffic profiles on the Cisco Nexus 92160YC-X and 9272Q switches: the more mice flows there are in the system, the bigger the overall performance gain will be.

As a conclusion, the testing results through the Cisco Nexus 92160YC-X and 9272Q switches validated that the algorithm-based intelligent buffering and scheduling approach address the real-world network congestion problems caused by traffic bursts more efficiently, and demonstrated overall better application performance in comparison to the deep buffer approach represented by the Arista 7280SE-72 switch in this test.

# 8 – References

Why flow-completion time is the right metric for congestion control
http://yuba.stanford.edu/techreports/TR05-HPNG-112102.pdf

Experimental Study of Router Buffer size
http://yuba.stanford.edu/~nickm/papers/IMC08_buffersizing.pdf

On the Data Path Performance of Leaf-Spine Datacenter Fabrics
https://people.csail.mit.edu/alizadeh/papers/hoti13.pdf

DCTCP: Efficient Packet Transport for the Commoditized Data Center
http://research.microsoft.com/pubs/121386/dctcp-public.pdf

VL2: A Scalable and Flexible Data Center Network
http://research.microsoft.com/pubs/80693/vl2-sigcomm09-final.pdf

PIE: A lightweight control scheme to address the bufferbloat problem
http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=6602305&abstractAccess=no&userType=inst

## 9 - About Miercom Testing

This report was sponsored by Cisco Systems, Inc.  The data was obtained completely and independently by Miercom engineers and lab-test staff as part of our performance verification testing.  Testing such as this is based on a methodology that is jointly co-developed with the sponsoring vendor.  The test cases are designed to focus on specific claims of the sponsoring vendor, and either validate or repudiate those claims.  The results are presented in a report such as this one, independently published by Miercom.

## 10 - About Miercom

Miercom has published hundreds of network-product-comparison analyses in leading trade periodicals and other publications. Miercom's reputation as the leading, independent product test center is undisputed.

Private test services available from Miercom include competitive product analyses, as well as individual product evaluations. Miercom offers comprehensive certification and test programs including: Certified Interoperable, Certified Reliable, Certified Secure and Certified Green. Products may also be evaluated under the Performance Verified program, the industry's most thorough and trusted assessment for product performance.

## 11 - Use of This Report

Every effort was made to ensure the accuracy of the data contained in this report but errors and/or oversights can occur.  The information documented in this report may also rely on various test tools, the accuracy of which is beyond our control.  Furthermore, the document relies on certain representations by the vendors that were reasonably verified by Miercom but beyond our control to verify to 100 percent certainty.

This document is provided "as is," by Miercom and gives no warranty, representation or undertaking, whether express or implied, and accepts no legal responsibility, whether direct or indirect, for the accuracy, completeness, usefulness or suitability of any information contained in this report.

No part of any document may be reproduced, in whole or in part, without the specific written permission of Miercom or Cisco Systems, Inc. All trademarks used in the document are owned by their respective owners. You agree not to use any trademark in or as the whole or part of your own trademarks in connection with any activities, products or services which are not ours, or in a manner which may be confusing, misleading or deceptive or in a manner that disparages us or our information, projects or developments.