BROADCOM®

White Paper

Introduction to Congestion Control for RoCE

by Moshe Voloshin, System Architect, Broadcom Inc.

Overview

By their nature, data communication networks create the potential for directing more traffic to a link than the link can transport. The instantaneous overfeeding of a link is called contention. Contention is resolved by momentarily storing (queuing) some contending traffic in network elements until it can eventually be sent. For example, if two 100 Gb links simultaneously deliver a packet destined to the same 100 Gb, one packet must be queued until the other is sent. Sustained overfeeding, possibly until the network element storage is exhausted, is called congestion.

Network Congestion can only be resolved by detecting it and reducing the data delivery rate. Congestion control (CC) schemes combine a variety of mechanisms to detect congestion and reduce the data rate in response.

There are well-known TCP Congestion Control algorithms, however, AI, HPC, and modern storage applications are driving adoption of RDMA protocols such as RoCE which require different CC solutions. Some aspects of the required CC modifications relate to protocol differences between RoCE and TCP and others relate to the significantly lower latency and Round Trip Delays in Data Centers employing RoCE transport.

Key figures of merit for congestion management:

- Utilization – Application data rate delivered from source to destination, also known as goodput.
- Waste – Transmission resources consumed in the operation of the congestion management scheme, such as congestion control packets, embedded congestion control information, and discarded data.
- Buffer consumption – The amount of memory consumed by transit data in network elements. Buffer consumption impacts application latency linearly as a result of queuing backlog and nonlinearly when total buffer consumption exceeds available switch buffer capacity. These impacts cause packet drops and associated slow-paced recovery or engagement of brute-force flow control.
- Latency – The time for application data to be delivered from source to destination.
- Fairness – The balance of service delivered across equally competing peers.

TCP combines its congestion management scheme and a byte-stream transport into a monolithic protocol.

The development of increasingly fast and increasingly dense networks over the past 20 years has created two tidal forces on TCP:

- A desire for even better performance in specific network scenarios; for example, dense, regularly structured networks with high-link bandwidths, short round-trip times, and minimally buffered participating network elements (as in modern data centers).
- A desire for transport models that lower application-to-application latency and increase endpoint data-handling efficiency.

The performance itch is being scratched with new congestion management schemes while Remote Direct Memory Access (RDMA) protocols offer low-latency kernel-bypassed messaging with highly efficient zero-copy transport.

This paper introduces RoCE, reviews TCP CC, describes current RoCE CC algorithms, and presents supporting data.

RDMA Over Converged Ethernet

RoCE supports three primary types of data exchange:

- Messages – RDMA send
- Remote memory access – RDMA write and RDMA read
- Atomic remote memory access – RDMA atomics

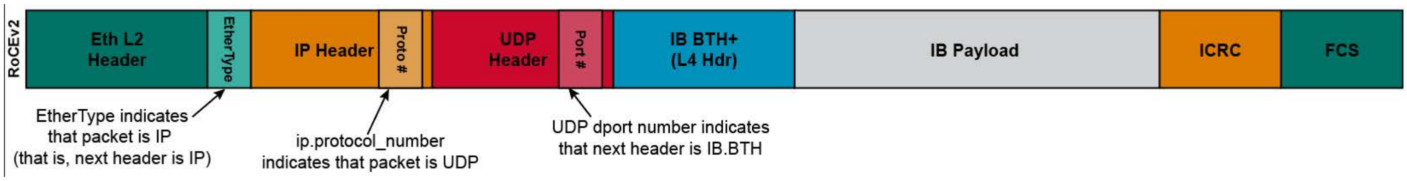
RoCE is defined with the specific goal of enabling highly efficient hardware implementations in RoCE NICs (RNICs). A debate has continued for decades over whether network transport protocols should be implemented in hardware. RNICs have proved an enduring technology, offering the following advantages:

- Zero-copy transfer among arbitrary, application-supplied buffers. The streaming nature of TCP requires applications to either accept intermediate buffers, or, more commonly, endure a data copy between the transport protocol layer and the application that incurs significant additional CPU consumption and latency.
- Kernel-bypassed operation. Kernel bypass allows user-mode applications to directly access the network interface, with low application-to-application latency.
- Extension of a memory (e.g. RAM) access model to the network. While a RAM model can be built on a CPU-based stream transport, hardware RNICs deliver a cost model low enough for remote memory access that existing RAM-based algorithms can be scaled-out efficiently.

RoCE's design is centered around point-to-point communications, wherein arbitrary-sized sends and RDMA reads/writes are transported by RNIC hardware, including segmentation, reassembly, reliability, sequencing, and error recovery.

RoCE is an adaptation of the InfiniBand (IB) system-area network RDMA protocol to Ethernet[1]. RoCEv2 mapped RoCE onto UDP/IP, as shown in Figure 1.

Figure 1: RoCEv2 Packet Formats


RoCEv2
Eth L2 Header
EtherType
IP Header
Proto #
UDP Header
Port #
IB BTH+ (L4 Hdr)
IB Payload
ICRC
FCS
EtherType indicates that packet is IP (that is, next header is IP)
ip.protocol_number indicates that packet is UDP
UDP dport number indicates that next header is IB.BTH

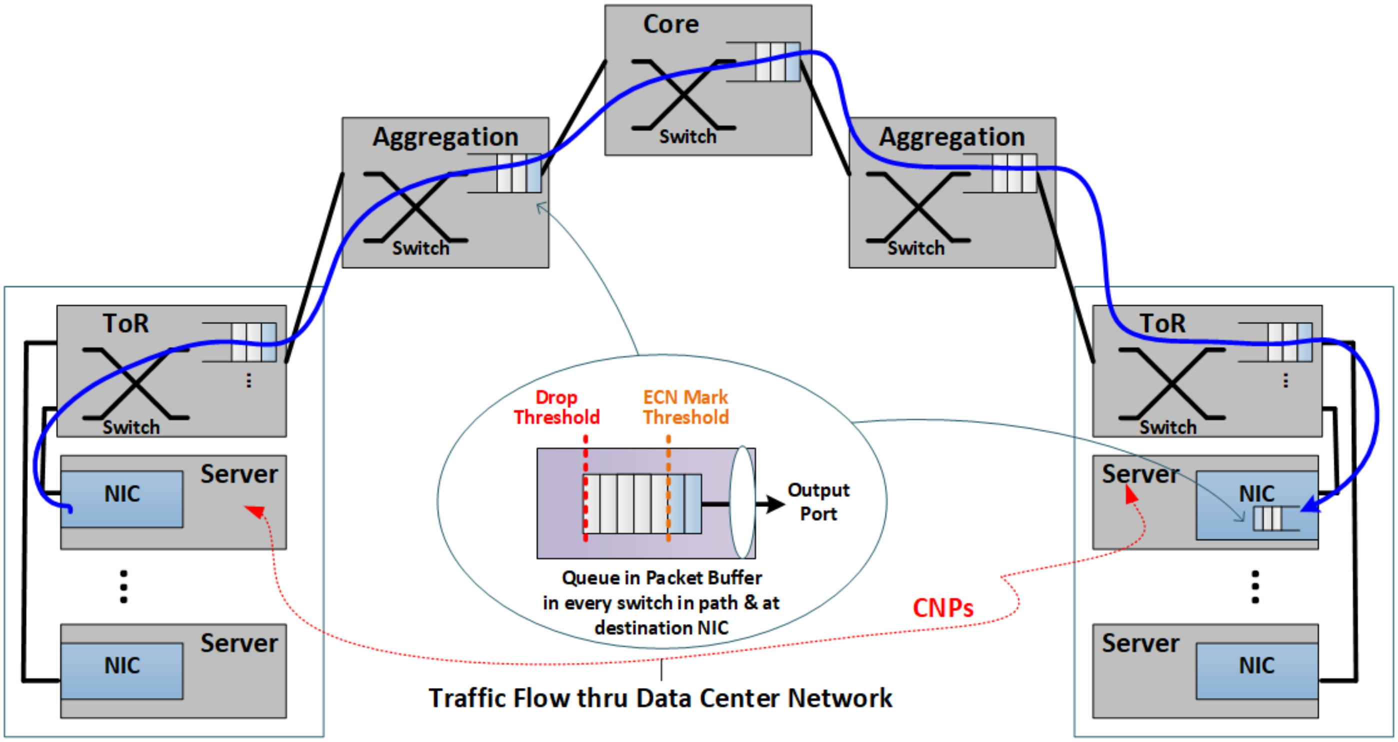
RoCEv2 provides three advantages:

- Operation on routed networks ubiquitous in large data centers
- IP QoS – The DiffServ code point (DSCP), or alternatively VLAN PRI
- IP congestion – The explicit congestion notification (ECN) signal

The intermediate UDP layer is only present to provide a connection identifier. The layer is a UDP/IP quintuple (da, sa, protocol, dport, sport) for existing hardware and software network management mechanisms, including ECMP- based load balancing and network instrumentation.

[1] InfiniBand Trade Association. InfiniBand™ Architecture Specification Release 1.2.1 Annex A17: RoCEv2. 2 September 2014.

Figure 2: Traffic Flow through the Data Center Network



RoCEv2 also defines a Congestion Notification Packet (CNP), shown in Table 1. RNICs send CNPs in response to ECN Congestion Experienced (CE) markings to indicate that the transmission rate should be reduced. ECN marking is done by switches along the path between source and destination or by the receiving NIC. CNPs are associated with RoCE connections, providing fine-grained, per-connection congestion notification information. RoCEv2 only specifies the mechanism for marking packets when congestion is experienced and the format of the CNP response. It leaves the particular congestion control algorithm unspecified, including the following information:

- When packets are ECN marked (at which queue level, and at what probability)
- When CNPs are generated in response to ECN
- How sending rate is adjusted in response to CNPs

Table 1: RoCEv2 CNP Format

IPv4/IPv6 Header
UDP Header
BTH DestQP set to QPN for which the RoCEv2 CNP is generated. Opcode set to b’10000001. PSN set to 0. SE set to 0. M set to 0. P_KEY set to the same value as in the BTH of the ECN packet marked.
(16 bytes) – Reserved. MUST be set to 0 by sender. Ignored by receiver.
ICRC
FCS

Ethernet flow control and priority flow control (PFC) are fully standardized mechanisms to prevent RoCE senders from overfeeding a network. Unfortunately, as explained below, the performance of Ethernet flow control alone on a loaded network is quite poor in every metric except waste.

Ethernet Priority Flow Control

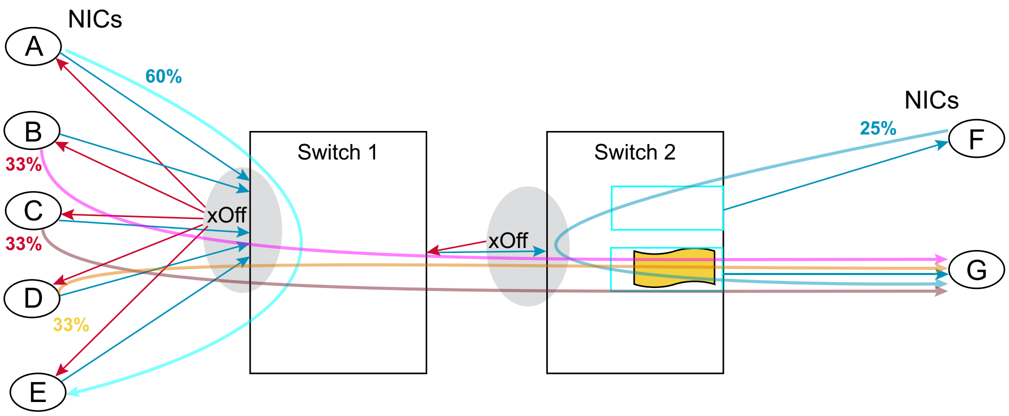
Link level pause and Priority Flow control (PFC) are mechanisms for restricting flow on a particular traffic class (up to eight traffic classes, associated with the eight Ethernet VLAN tag priorities or with DSCP) on a link[2]. Careful configuration of flow control thresholds can ensure no packet drops across a link. In other words, PFC can be used to implement lossless traffic classes. This, in turn, prevents endpoints from overfeeding the network.

The technical challenge in using PFC is that, when active, it restrains all traffic for the class whether or not it is destined for an overfed path. Furthermore, a single overfed path can lead to traffic being restrained on all network paths. This is called congestion spreading.

Figure 2 provides a simple example of congestion spreading:

1. NICs B, C, D, and F overfeed NIC G.
2. The queue to NIC G fills.
3. Switch 2 stops flow on all ports to prevent admission of packets potentially destined for NIC G.
4. The queue from Switch 1 to Switch 2 fills.
5. Switch 1 stops flow on all ports to prevent admission of packets potentially destined to Switch 2.
6. The path from NIC A to NIC E becomes blocked even though it is not overfed.

Figure 3: PFC Congestion Spreading



PFC degrades network performance in multiple ways:

- The latency through filled queues is very high
- Flow control turnaround latency compounds the already high queuing delay
- Network utilization plummets and fairness evaporates as uncongested flows become victims of congested flows

[2] IEEE. 802.11Qbb. Priority-based flow control, 2011.

Modern switches implement complex buffer and flow-control strategies to mitigate the impact of filled queues and congestion spreading, but all schemes eventually succumb at sustained high network load.

PFC-controlled networks only thrive at low sustained loads. In effect, PFC only provides a safety net to prevent packet loss that would precipitate high-cost recovery events. PFC cannot effectively contain network latency nor can it deliver high utilization.

Data Center TCP

Modern data center switches, such as those built on Broadcom's Trident and Tomahawk® family of chips, offer tremendous bandwidth density but with relatively modest buffering. Data Center TCP (DCTCP) is a stream transport protocol with a congestion management scheme that provides utilization and efficiency similar to TCP but with very little buffer consumption. This means DCTCP exhibits much lower steady-state latency and eliminates latency spikes from flow control or packet drops when switch buffers are exhausted.

DCTCP preserves the TCP method of controlling the transmit window. The transmit window size is the amount of unacknowledged data allowed in the network. When the transmit window is at its minimum, one segment (~1500 bytes) of data is sent per network round-trip-time (RTT). Note that this RTT includes the endpoint processing as well as queuing and transmission delays. When the window grows to bandwidth × RTT of the path, the full bandwidth of the path is used.

The transmit window mechanism was chosen in the initial invention of TCP to relieve endpoints of the substantial CPU burden of maintaining a transmit timer for every connection. TCP is self-timed— the network path itself is the timer.

The DCTCP congestion control scheme is:

- Network elements mark packets added to queues longer than a fixed, configured threshold. This differs from traditional TCP active queue management (AQM):
  - Instantaneous queue length is used, where traditional AQM uses average queue length.
  - Every packet of an overfilled queue is marked, where traditional AQM marks packets probabilistically as a function of average queue length.
- DCTCP receivers send an ECN-Echo indication for every ECN CE packet (subject to delayed ACKs). Traditional AQM sends ECN-Echo indicators in response to a single ECN CE until the sender acknowledges the ECN-Echo. ECN CE marking is expected to be a rare event in traditional AQM, whereas it is common in DCTCP.
- DCTCP senders increase transmit window size in the same way as TCP (slow-start and additive increase)
- DCTCP senders use a moving average of the ratio of ECN-Echo indications received to packets sent to approximate the probability congestion experienced, CP, and reduce transmit window size by (1 – CP / 2), thus DCTCP reduction could be more graceful than traditional TCP.

DCTCPs mechanisms ensure the window is not reduced when no congestion is experienced, and the window is cut in half when all packets experience congestion, as in TCP. DCTCP reduces the current window gingerly and continuously between these two extremes, quite unlike TCP.

DCTCP's immediate, universal congestion notification ensures all congesting senders are notified and react extremely quickly. This is the core property that makes DCTCP so effective in minimizing buffer consumption.

Congestion Control for RoCE: DCQCN

The key differences between TCP and RoCE are:

- TCP is stream-based while RoCE is message-based
- TCP implementations are typically software-based while RoCE is implemented in the hardware
- TCP controls an inflight window, number of unacknowledged bytes, while RoCE controls the transmission rate

Broadcom Ethernet NICs support two CC modes, DCQCN-p and DCQCN-d, where DSQCN-p utilizes probabilistic ECN marking policy, with marking probability increasing linearly within a range of congested queue levels, while DCQCN-d utilizes Deterministic ECN marking policy as in DCTCP where 100% of the packets are marked when congested queue level rise above a configured threshold.

In both modes, the NIC performs very similar operations and utilizes the same infrastructure to control the rate of each flow (Queue Pair, or QP, in RoCE terminology). But since the number of ECN marked packets and hence CNPs differ, the computation of congestion level is different.

In DCQCN-p there are fewer CNPs than in DCQCN-d since when congested queue level starts to rise, only a small percentage of packets traversing the switch are ECN marked. Some of the flows which do receive CNPs reduces their rate while other do not. If congestion still persists, higher percentage of packets are marked and more flow possibly receive a signal from the network and reduce their rate. Thus when there are large number of competing flows, the congested queue level may rise to higher level until stabilizing in comparison with DCQCN-d. On the other hand, since there are more CNPs with DCQCN-d, there is a higher load on the NIC in processing the stream of CNPs and accessing the associated flow context.

The CC algorithm in Broadcom Ethernet NICs has been enhanced relative to the original DCQN paper[3] due to several issues in the original algorithm.

Several papers have been published about deficiencies of DCQCN, among them are:

“Y. Gao, Y. Yang, T. Chen, J. Zheng, B. Mao, and G. Chen. Dcqcn+: Taming large-scale incast congestion in rdma over Ethernet networks. In 2018 IEEE 26th International Conference on Network Protocols (ICNP), pages 110–120, Sep. 2018”

RoCE CC, like most congestion control schemes, involves two separate rate control mechanisms: one to increase the rate when no congestion is experienced and another to reduce the rate in response to congestion.

CC rate management is expressed in terms of two variables:

- RC – Current transmit rate
- RT – Target transmit rate

These variables are updated periodically based on previous values, congestion indications, and internal algorithmic state. The CC update period is configurable, with default being 45µs. The behavior is relatively insensitive to moderate variations of this period. During each control loop period the transmitting NIC tracks the number of transmitted packets and the number of received CNPs on each flow and adjust the flow’s transmission rate accordingly. Broadcom Ethernet NICs offload the complete RoCE transport operation to HW, including per-flow Congestion state management.

[3] Y. Zhu, H. Eran, D. Firestone, C. Guo, M. Lipshteyn, Y. Liron, J. Padhye, S. Raindel, M. H. Yahia, and M. Zhang, “Congestion control for largescale rdma deployments,” Acm Sigcomm Computer Communication Review, vol. 45, no. 4, pp. 523–536, 2015.

DCQCN Rate Increase

DCQCN builds on pure rate control rather than a transmit window, and leverages IEEE 802.1Qau Quantized Congestion Notification (QCN) scheme[4].

Figure 3 illustrates DCQCN's rate increase mechanism. The DCQCN rate increase phase can be understood as starting after receiving a CNP and continuing until it captures full link BW or until it the next CNP is received, indicating congestion. The DCQCN rate increase has two phases:

- Fast Recovery – Rapidly increasing RC towards the RT captured before the last CNP was received by periodically halving the distance to RT:

  RC = (RT + RC) / 2

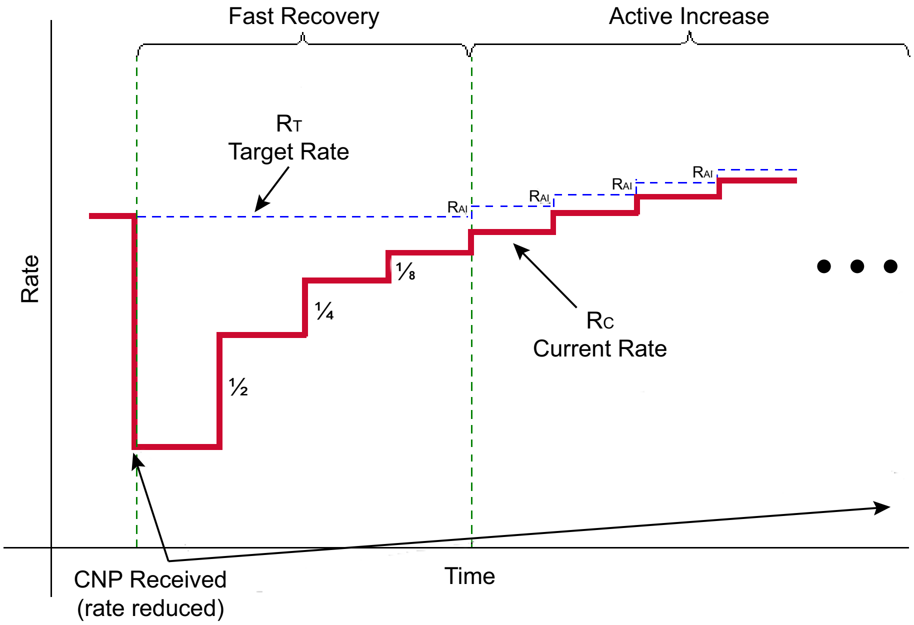
  DCQCN performs a configurable number of Fast Recovery steps, three by default, before switching to Active Increase. This phase is important to maintain high link utilization in steady-state.

- Active Increase – Periodically increasing RT by a fixed minimum increment RAI and increasing RC by half the distance to RT until congestion is reported or the full rate of the link is achieved:

  RT = RT + RAI

  RC = (RT + RC) / 2

Figure 4: NDCQCN Rate Increase Diagram



These rules ensure DCQCN quickly approaches a new set point after congestion is reported, and constantly probes for additional bandwidth after recovery.

[4] IEEE. 802.11Qau. Congestion notification, 2010.

DCQCN Rate Decrease

When CNPs are received, the rate is reduced by an amount proportional to an estimated congestion probability (CP). Here resides the differences between DCQCN-p and DCQCN-d. With deterministic marking CNPs are received more often and thus a sending flow may receive several CNPs in each control loop period. This is in essence a multibit congestion signal which gives a measure of congestion vs. just the presence of temporary congestion. Therefore, to compute the running estimate of congestion probability, the ratio between the number of CNPs received and number of packets transmitted each period is computed and would result in a value between zero and one. If the ratio is high, and repeating for several periods, CP will keep rising. CP then is used to determine the level of rate reduction the sending flow will perform in each period it receives a CNP.

With probabilistic marking, since the number of CNPs expected to be received is much smaller, the signal just indicates the presence congestion, thus the computed ratio for fraction of CNPs (F in equations below) is set to one when a CNP was received during the control loop period and to 0 otherwise. Each time a CNP is received CP will rise by approximately the weight g times the ratio F. The weight is configurable in both modes. The higher the weight the quicker CP will increase and the shorter the ‘memory’ of congestion is.:

$F = \text{No. of CNPs} / \text{No. of TX messages}$ $CP = (1 - g) \times CP + F \times g$

$RT = RC$

$RC = RC \times (1 - CP / 2)$

The intuition behind these equations:

- CP is a moving average of the probability a packet experiences congestion.
- The target rate is set to the current rate when congestion is experienced.
- If CP is minimal (one packet experiences congestion), the rate is minimally reduced.
- If CP is maximal (all packets experience congestion), the rate is halved.

Enhancements in DCQCN

A number of issues in the original DCQCN paper have been identified and investigated over the past several years. Broadcom Ethernet NICs are enhanced to mitigate many of these problems.

The reaction Point (RP) utilizes existing state variables such as current rate (cr), the time between transmissions on each QP, and time between reception of CNPs to extract additional information about the scale of congestion and the present level of incast. The RP then uses this additional information to adjust how often a QP increases rate in absence of congestion instead of increasing rate every fixed period of time to reduce the likelihood of building large queue in switches under large incast and thus reduce the end-to-end latency through a congested network.

The RP in Broadcom Ethernet NICs also adds random jitter to control loop period to avoid synchronization in reaction among many flows. This feature prevents potential synchronized start stop among flows which could cause large bursts to be transmitted as well as significant quiet time (no transmission). The first could cause abrupt buildup of congested queue level and the second could lead to gaps in transmission and under-utilization of network BW.

It is well known that HyperActive increase is not an effective measure to provide quick BW ramp up after congestion ceases. This is pointed out in the original DCQCN paper. Broadcom Ethernet NICs employ instead a super linear rate increase in absence of congestion which allows the flows which remain active after congestion cease to capture the available BW very quickly. The longer the duration of no CNPs the higher the rate increase is. Thus, after congestion cease and there is one or only few remaining active QPs, each QP increases transmission rate slowly at first. With each additional period with no congestion each QP doubles the magnitude of rate increase and hence ramps up to the maximum possible BW significantly faster than if it uses a fixed small rate increase.

These enhancements and other optimizations allow Broadcom Ethernet NIC's CC algorithm to maintain a low queue level which in turn reduces the end-to-end loaded latency and avoids using PFC to mitigate large incast.

All RoCE CC functionality is performed in hardware. No software nor firmware is involved in the control loop. Therefore, Broadcom Ethernet NICs can keep up with high rate of congestion signal from the network when large number of flows compete for BW, and maintain real-time accurate state of congestion per QP at all times and under all conditions.

Summary

DCQCN is a high-performance congestion management solution for RoCE that delivers fairness, high utilization, dramatically lower buffer consumption and network latency, and provides connection scaling. DCQCN ensures Broadcom's Ethernet NIC family of products deliver the best possible RoCE performance, and especially the lowest possible end-to-end latency, in the industry. [5] [6]

[5] Y. Zhu, H. Eran, D. Firestone, C. Guo, M. Lipshteyn, Y. Liron, J. Padhye, S. Raindel, M. H. Yahia, and M. Zhang. Congestion Control for Large-Scale RDMA Deployments. In SIGCOMM, 2015.

[6] C. Guo, H. Wu, Z. Deng, G. Soni, J. Ye, J. Padhye, M. Lipshteyn. RDMA over Commodity Ethernet at Scale. In SIGCOMM, 2016.

Broadcom reserves the right to make changes without further notice to any products or data herein to improve reliability, function, or design. Information furnished by Broadcom is believed to be accurate and reliable. However, Broadcom does not assume any liability arising out of the application or use of this information, nor the application or use of any product or circuit described herein, neither does it convey any license under its patent rights nor the rights of others.