aruba
a Hewlett Packard Enterprise company

VRD

RDMA OVER CONVERGED ETHERNET (ROCE) DESIGN GUIDE

REDUCING LATENCY, CPU OVERHEAD, AND ENABLING FAST TRANSFER OF DATA

Introduction

This guide provides information and use cases on Aruba Data Center Bridging (DCB) solutions for environments that leverage Remote Direct Memory Access over Converged Ethernet (RoCEv1/v2) solutions.

Modern data center challenges—what is the problem?

Within today's enterprise, servers are required to handle massive amounts of data while providing 100% uptime. Over the years, adoption of server virtualization, big data analytics and the proliferation of mobile devices have continued to stress computing infrastructures. Users have noticed applications taking longer than they should to execute. When corporate and other users notice slowing of the systems, they become less productive. Many times, this type of delay happens because large amounts of data has to be processed by the CPU which then has to move from buffer spaces, down through the TCP stack, onto the wire between servers of the enterprise, and then back up the stack again on the other side. This transfer can cause the CPU to slow down processing of other tasks as the CPU is busy. Adding more servers may increase CPU processing power but it is not addressing the fact that the CPUs are getting over-utilized and it runs counter to the goal of doing more with less within today's organizations.

Remote Direct Memory Access (RDMA) Solution

RDMA enables the movement of data between servers with very little CPU involvement.

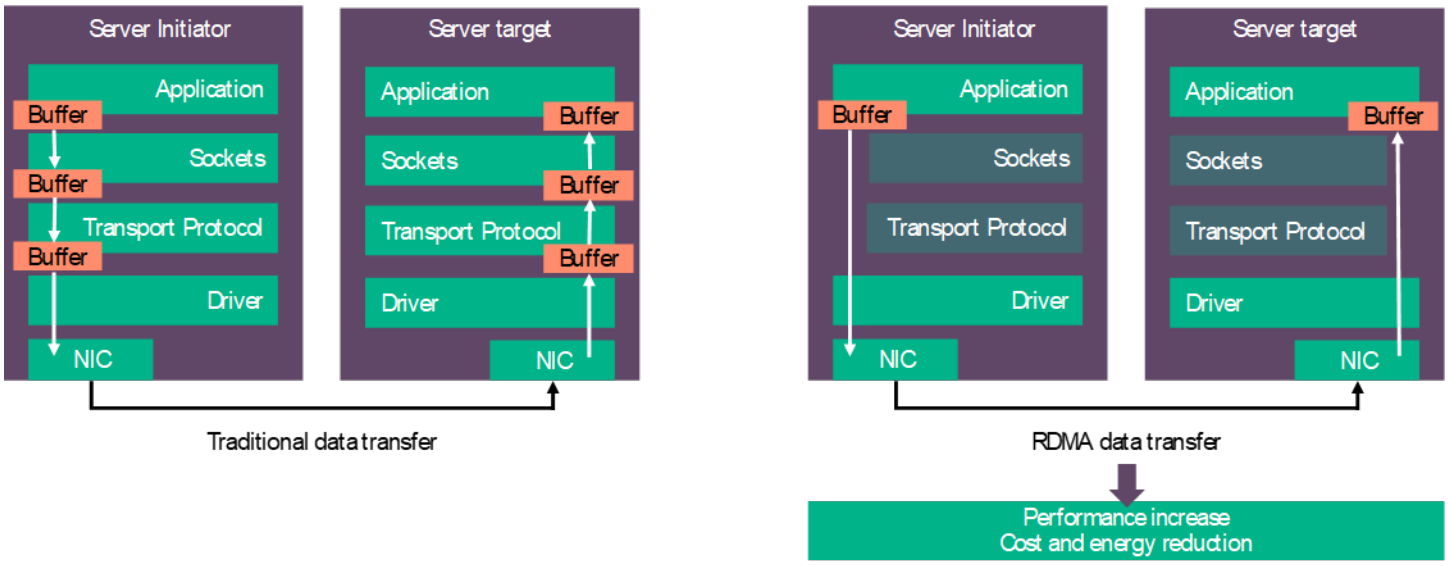


Figure 1. RDMA bypassing OS stack

Without RDMA, traditional movement of data utilizes TCP/IP buffer copies and significant overhead. Applications reply on the OS stack to move data from memory, virtual buffers through the stack, onto the wire, across the wire, and then again back up the wire. The receiving OS must retrieve the data and place it directly in the application(s)' virtual buffer space which leads to the CPU being occupied for the entire duration of read and write operations and is unavailable to perform other work.

RDMA solutions are able to bypass the OS stack. The OS is used to just establish a channel which applications use to directly exchange messages on. A network adapter transfers data directly to and from application memory eliminating the need to copy data between application memory and the data buffers within the operating system. Such communication requires no work to be done by CPUs, caches or context switches, and transfers continue in parallel with other system operations. When an application performs an RDMA Read or Write request, the application data is delivered directly to the network, reducing latency, CPU overhead, and enabling fast transfer of data.

Users running RDMA applications on an Ethernet network can see application performance improvements that derive from the offloading of data movement and higher availability of CPU resources to applications. Shifting the chore of data movement from the CPU makes both data movement and execution of applications more efficient. RDMA delivers performance and efficiency gains that are not available from any other communications protocol; including low latency, improved resource utilization, flexible resource allocation, fabric unification and scalability. Greater server productivity lowers the need for additional servers and lowers the total cost of ownership.

- RDMA is the direct read from or write to an application's memory
- Hardware offload moves data faster with significantly less overhead allowing the CPU to work on other applications
- CPU initiates the transfer and processes other operations while the transfer is in progress
- Ultra–low latency through stack bypass and copy avoidance
- Reduces CPU utilization
- Reduces memory bandwidth bottlenecks
- Enables high bandwidth and I/O utilization
- RDMA is useful when CPU cannot keep up and needs to perform other useful work

RDMA transport technologies

There are three main transport types of solutions that can be used to transport RDMA over an Ethernet network.

- InfiniBand (IB)
  - Protocol which supports RDMA natively from the beginning
  - Requires dedicated NICs and switches that supports this technology
  - Pure InfiniBand solutions can provide high performance at cost of dual networking fabrics
- Internet Wide Area RDMA Protocol (iWARP)
  - RDMA over TCP
  - iWARP defined by IETF and uses the TCP/IP stack in order to be compatible with any Ethernet/IP infrastructure
  - Data Center Bridging (DCB) Ethernet helps avoid congestion, but it is not required by the standard
  - Supports offload to the NIC
  - Goes up the TCP/IP stack to achieve protection for loss
- RDMA Over Converged Ethernet (RoCE)
  - Data Center Bridging (DCB) Ethernet should be configured, but it is not required by the standard
  - Requires a DCB switch to provide for a lossless fabric
  - NICs should support RoCE and offloading
  - Lower level Ethernet mechanisms used to protect for loss:
    - Priority Flow Control (PFC) to stave off loss
    - Enhanced transmission selection (ETS) to protect traffic classes (TC)
  - Uses upper InfiniBand layers in case of need for retransmission to recover from loss.

The Aruba CX solutions can support the RoCE based version and therefore the following content will focus on RoCE based networking.

What is RoCE?

RoCE is a network protocol that allows RDMA over Converged Ethernet, or RoCE (pronounced "rocky"). This critical technology is now expanding into enterprise markets where Ethernet networks are ubiquitous. RoCE is geared for high performance within an advanced data

center architecture eliminating dedicated storage area networks (SANs) by converging compute, network, and storage onto a single fabric. Utilizing advanced reliable Ethernet and DCB with RDMA techniques, RoCE provides lower CPU overhead and increases enterprise data center application performance.

Today's dynamic evolving enterprise, let it be local, remote cloud, or hybrid data centers, require high performance technologies like RoCE to support increasingly data intensive applications and the move to hyper converged scale-out solutions which leverage distributed computing/storage models.

Aruba Networks currently supports RoCE solutions using the CX 8325/8360/8400 Switches.

The benefits of implementing RoCE are:

- Lower cost of ownership
- Greater return on investment that span traditional and today's hyper converged infrastructures
- Reduces CPU over utilization
- Reduces Host Memory Bottlenecks
- Helps to better leverage the storage media evolution which has brought 10,000x performance improvement factor
- Offloads memory access process
- Increases throughput and lowers latency between compute and storage systems

RoCE aspects

The initial RoCE v1 solution simply replaced the IB Link Layer with an Ethernet link layer. In this solution RoCE was a Layer 2 based Ethernet solution. The latest version of RoCE, which is called RoCE v2, replaced the IB Network layer with a standard IP and UDP Header so traffic is routable now.

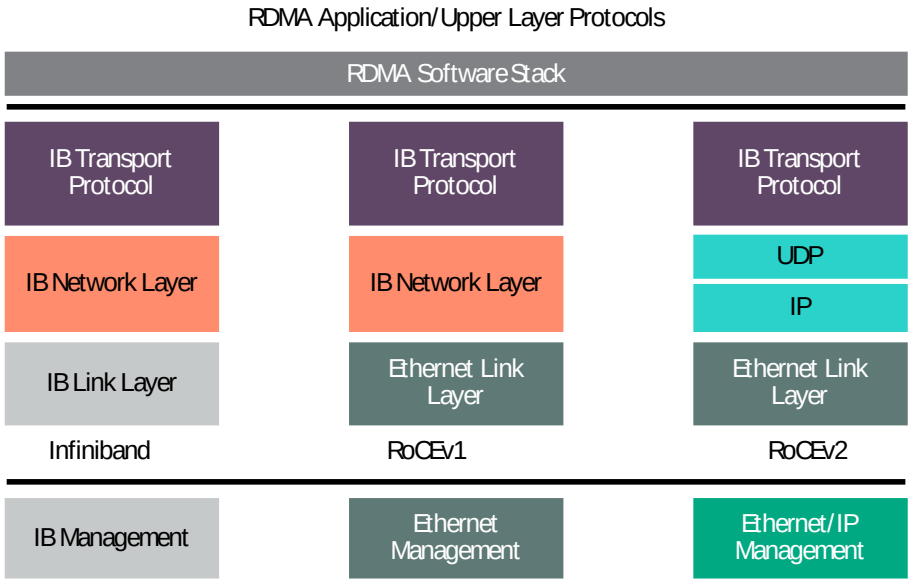


Figure 2. IB/RoCE evolution

The InfiniBand Annexes for RoCEv1 and RoCEv2 do not actually mandate that no loss occur on the Ethernet network. However, even though not mandatory or required by the standard, RoCE based solutions will perform poorly in lossy fabrics because in times of congestion devices will start to drop packets, causing retransmissions, reducing throughput, and increasing delay. In these situations, the solution will simply not get the RDMA benefits needed to reduce CPU load and latency.

Aruba Networks recommends, in production environments, RoCE based solutions should be deployed as a lossless fabric.

RoCE use cases

RDMA and RoCE have seen growing market adoption. Below is a summary of the markets which have started to leverage these solutions.

Cloud computing

The cloud computing market has been actively leveraging the benefits of RDMA and RoCE. These environments obtain benefits, such as improved SLAs through deterministic performance, efficient clustering allowing for elastic/scale out computing. As indicated before, the benefits of implementing RoCE are a lower cost of ownership and greater return on investment that span traditional and modern hyper converged infrastructures.

These environments are able to use the following applications (and more) that leverage RDMA/RoCE:

- VMware®
  - VMware has published Ultra-Low Latency on vSphere with RDMA which compares RDMA vs regular TCP/IP stack in cloud environment showing following benefits:
    - Total vMotion traffic time is 36% faster.
    - 30% higher copy bandwidth ability.
    - CPU utilization is from 84% to 92% lower.
- Microsoft® Azure
  - Video describing how a 40GbE implementation RoCE solution provided 40Gbps throughput at 0% CPU utilization with Azure
- Red Hat® KVM
- Citrix® Xen
- Microsoft®
- Amazon EC2
- Google™ App Engine

Data storage

Many data storage focused applications are gaining benefits from implementing RDMA/RoCE. A couple of examples are Microsoft SMB Direct, and Lustre. Data Storage protocols over RDMA deliver higher throughput and lowers latency.

Everyday application like Exchange, Lync® and SharePoint are also able to leverage and benefit from the use of RoCE.

Financial services

Financial Service market has been leveraging InfiniBand for some time now as those environments have a high demand for low latency. The following are applications which could see high performance and I/O benefits in RoCE solutions:

- Tibco
- Wombat/NYSE
- IBM WebSphere MQ
- Red Hat MRG
- 29West/informica

Web 2.0 big data

Big data environment is another segment which can benefit from RoCE solutions. The goal in these big data environments is to minimize response time and increase I/O.

The following are applications which could see benefits in RoCE solutions:

- Storage Spaces Direct (S2D):
  - Software-defined storage (SDS) stack in Windows Server which enables building highly-available (HA) storage systems with local storage
- SMB 3 Direct:
  - An extension of Microsoft Server Message Block technology used for file operations. Direct implies use of high speed Remote Data Memory Access (RDMA) networking methods to transfer large amounts of data with little CPU intervention
- Multiple other workload types can benefit (Big Data, Databases, DFS, Cloud):
  - MicroSoft Azure Stack HCI via RoCEv2
  - Open Source Lustre
  - Oracle RAC
  - IBM DB2 PureScale
  - MS SQL Server
  - Hadoop
  - Eucalyptus
  - Cassandra

RoCE design recommendations

Lossless network fabrics

One of the objectives when designing network solutions that incorporate RoCE is to deploy a lossless fabric. Even though the RoCE standards do not necessarily demand lossless networks they will perform much better when deployed in a lossless manner, especially under heavy congestion. With that in mind, Aruba Networks recommends that a lossless fabric be considered a requirement for RoCE implementations.

Lossless fabrics can be built on Ethernet fabrics by leveraging the following DCB protocols.

- Priority- based flow control (PFC): IEEE standard 802.1Qbb, is a link-level flow control mechanism. The flow control mechanism is similar to that used by IEEE 802.3x Ethernet PAUSE, but it operates on individual priorities. Instead of pausing all traffic on a link, PFC allows you to selectively pause traffic according to its class.
- Enhanced Transmission Selection (ETS): ETS helps to ensure that when using PFC, lossless traffic does not get continually paused because other types of traffic are using the whole bandwidth of a link. With ETS you can tell an interface that a certain percentage of the bandwidth is guaranteed for each traffic type. This way each traffic type gets a minimum amount of bandwidth.
- Data Center Bridging Exchange (DCBX): This protocol helps to ensure that the NIC and the Switch are configured properly. DCBx is a discovery and capability exchange protocol which discover peers and exchanges configuration information. The protocol allows auto exchange of Ethernet parameters and discovery functions between switches and end-points. If the server NIC has the DCBx willing bit turned on then after you configure the switch with the needed DCB and traffic marking rules, then DCBx will ensure the server NIC also knows how to mark and treat traffic on that link.
- Quantized Congestion Notification (QCN): This protocol provides a means for a switch to notify a source that there is congestion on the network. The source will then reduce the flow of traffic. This help to keep the critical traffic flowing while also reducing the need for pauses. This is only supported in pure Layer 2 environments and seen very rarely now that RoCEv2 is the predominant RoCE solution.

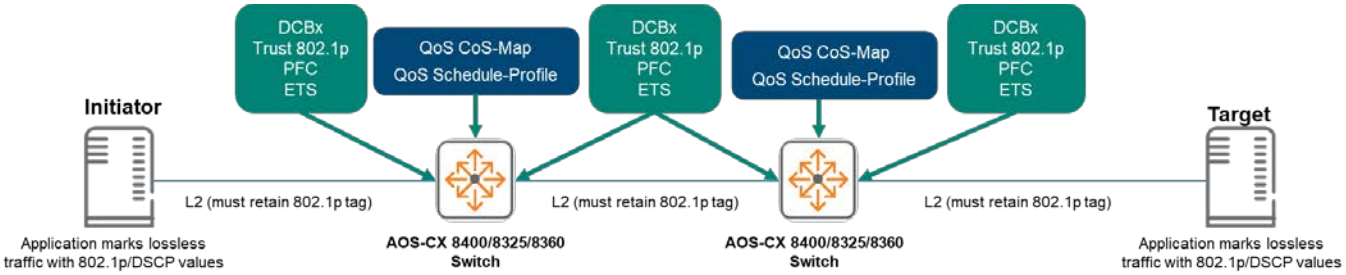
- IP Explicit Congestion Notification (IP ECN): IP explicit congestion notification is not officially part of the DCB protocol suite, however, it can be leveraged to enable end-to-end congestion notification between two endpoints on TCP/IP based networks. The two endpoints are an ECN-enabler sender and an ECN-enabler receiver. ECN must be enabled on both endpoints and on all the intermediate devices between endpoints for ECN to work properly. ECN notifies networks about congestion with the goal of reducing packet loss and delay by marking the sending device to decrease the transmission rate until congestion clears, without pausing packets.

Note:
IP ECN is not specifically a DCB standard, but it must be used in RoCEv2 solutions (see details later).

As shown in the image below, the hosts (initiators/targets) will be configured so that the lossless traffic flows are sent to the attached switch with the proper 802.1P/DSCP values.

The RoCEv1 configuration on the switches would apply Queue-Profiles to each switch to ensure those marked packets are in the proper queue. A Queue-Schedule would be created on each switch if any of the links was carrying both lossless and lossy traffic. The Queue-Schedule will get applied to each interface that carries both types of traffic. PFC will be applied to the needed interface/queues to ensure traffic in the proper queue do not get dropped. DCBx LLDP should be turned on and Trust 802.1P should be applied to all interfaces in path carrying lossless traffic.

Finally, the DCBx Application TLV can be used to tell the attached host to send lossless traffic marked with the proper 802.1p code point. This helps ensure the switch treats the important traffic properly. If the attached host does not have the willing bit turned on then the admins will need to manually configure the Hypervisor to mark traffic properly.



RoCEv1 Configuration guidance and details

Below are steps that should be taken when configuring RoCEv1 solutions with Aruba CX:

1. Enable LLDP and DCBx
2. Configure QoS Queue-Profile (2 queue profile allows for best absorption – lossy traffic in 1 queue / lossless traffic in other queue)
3. Configure Global Trust
4. Configure QoS Schedule Profile (in converged environments)
5. Configure PFC on interface/queus that require lossless Ethernet
6. Configure Application TLV (if hosts support DCBx)

Configure LLDP and DCBx

DCBx is a discovery and capability exchange protocol which exchanges configuration information between attached devices. The exchanged info can be useful in troubleshooting mismatches between endpoints.

LLDP DCBx can be enabled either globally or on an interface level.

Step	Command
Enable LLDP (enabled by default) ("no" form disables)	`switch(config)# lldp`
Verify LLDP status	`switch(config)# show lldp configuration`
Enable DCBx Globally (disabled by default) ("no" form disables)	`switch(config)# lldp dcbx`
Enabling DCBx on an interface. (enabled by default once enabled globally).	`switch(config-if)# lldp dcbx`
Verify DCBx status	`switch(config)# show dcbx <interface>`

See the Aruba CX Fundamentals Guide for more details on LLDP.

Configure QoS Queue-Profile

The next step involves creating a queue profile which ensures that 802.1p marked traffic gets placed into the proper queues. For example, each switch has 8 queues, but if we need to treat some traffic with PFC we should consider changing the queuing scheme (using the queue-profile) to a two queue model. In this model, we place all lossy traffic in one queue, while all lossless traffic gets placed into the other queue. This allows for better burst absorption.

Below is a simple 2-queue configuration example on Aruba CX switches. In this example, all 802.1p 4 traffic will be placed into queue 1, while all other traffic will get placed into queue 0.

Step	Command

Configure QoS Queue Profile	`switch(config)# qos queue-profile SMB`
	`switch(config-queue)# map queue 0 local-priority 0`
	`switch(config-queue)# map queue 0 local-priority 1`
	`switch(config-queue)# map queue 0 local-priority 2`
	`switch(config-queue)# map queue 0 local-priority 3`
	`switch(config-queue)# map queue 1 local-priority 4`
	`switch(config-queue)# map queue 0 local-priority 5`
	`switch(config-queue)# map queue 0 local-priority 6`
	`switch(config-queue)# map queue 0 local-priority 7`

Verify QoS Queue Profile	`switch(config)# show qos queue-profile SMB`
Apply QoS Queue Profile	`switch(config-if)# apply qos queue-profile SMB`

See the Aruba CX QoS Guide for more details on queue-profiles.

Configure Global Trust

We need to ensure that the proper trust configurations are applied to the proper ports. With RoCE based solutions that largely rely on the 802.1p marking we need to ensure that those marking are being trusted properly.

A best practice would be to set the global trust mode to CoS. This will be applied to all interfaces that do not already have an individual trust mode configured. A DSCP override can then be applied to any Layer 3 interfaces that do not carry the 802.1p tag.

Step	Command
Configure Global Trust	`switch(config)# qos trust cos`

Verify Global Trust Mode	`switch(config)# show qos trust`

See the Aruba CX QoS Guide for more details.

Configure QoS Schedule Profile

A schedule profile must be always defined on all interfaces and each port can have its own schedule profile. The schedule profile determines the order in which queues transmit a packet and the amount of service defined for each queue.

The objective of a RoCE based Schedule Profile is to ensure that the lossless traffic has enough bandwidth based on average bandwidth consumption of the port. There is no one size fits all for these ETS settings so this parameter may end up getting modified as the admins tune the setting for the specific environment.

The switch is automatically provisioned with a schedule profile named factory-default, which assigns WFQ/DWRR to all queues with a weight of 1. The default profile named "factory-default" is applied to all interfaces as well as a predefined profile named "strict." The strict profile uses the strict priority algorithm to service all queues of the queue profile to which you apply it.

There are three permitted configurations for a schedule profile:

1. All queues use the same scheduling algorithm (i.e., WFQ).
2. All queues use strict priority.
3. The highest queue number uses strict priority, and all the remaining (lower) queues use the same algorithm (i.e., WFQ).

Only limited changes can be made to an applied schedule profile. Any other changes will result in an unusable schedule profile, and the switch will revert to the factory default profile until the profile is corrected:

1. The weight of a dwrr queue.
2. The bandwidth of a strict queue.
3. The algorithm of the highest numbered queue can be swapped between dwrr and strict, and vice versa.

The below example shows an ETS configuration within a 2-queue environment. The settings use a weight to set the amount of available bandwidth for each queue. The settings in the example below would ensure that 50% of bandwidth will be applied to both queue 0 and queue 1.

Step	Command
Create a new schedule-profile called test SMB (no form removes)	`switch(config)# qos schedule-profile SMB1`

Configure each queue with appropriate bandwidth/algorithm.
 	
Example shown applies 50% to each queue
	`switch(config)# dwrr queue 0 weight 15`
	
	`switch(config)# dwrr queue 1 weight 15`

Verify schedule-profile	`switch(config)# show qos schedule-profile SMB1`

Apply schedule-profile	`switch(config-if)# apply qos schedule-profile SMB1`

See the Aruba CX QoS Guide for more details on Schedule Profiles.

Configure Priority Flow Control (PFC)

PFC enables flow control over a unified 802.3 Ethernet media interface, for local area network (LAN) and storage area network (SAN) technologies. PFC is intended to eliminate packet loss due to congestion on the network link. This allows loss sensitive protocols, such as RoCE to coexist with traditional loss-insensitive protocols over the same unified fabric.

Before the development of PFC a global pause was used which is port based. This means that in times of congestion all traffic on that port would get paused. PFC is port and queue based, which means it allows us to pause only traffic in a certain queue on a port. This way PFC simply pauses traffic that is in a lossless queue, so that no packets get dropped due to buffer congestion. All other traffic that is in the other queues on the port are not included for this type of checking and may be dropped as congestion occurs.

If you need to ensure that zero packets get dropped, then PFC must be deployed.

PFC details:

- Must be enabled on all endpoints and switches in the flow path
- Enables pause per hardware queue on an Ethernet device
- PFC is port and queue based (not flow based)
- Uses 802.1p CoS "Class of Service" values in 802.1Q VLAN tag to differentiate up to eight levels of CoS
- On L3 interfaces PFC requires preservation of 802.1Q tags
- On L3 interfaces if 802.1Q tags are not possible then traffic needs to be remarked to DSCP values
- Pause frames propagate hop-by-hop, without knowledge of the flows that are causing the congestion.
- To enable PFC at the interface level, DCBx must first be enabled.
- You can only configure 1 PFC priority per interface on 8325/8400
- You can configure 2 PFC priority per interface on 8360
- For the CX 8325 a reboot is required to enable PFC on the first interface: Subsequently, PFC can be enabled on more interfaces as long as they had never previously linked up since boot (i.e. dark ports).

When we configure PFC the idea is to configure PFC on an interface so that it knows not to drop traffic marked with a specific CoS value. By this point the queue-profile to be used should be applied, so admins should just need to enable PFC for the proper code points as needed.

The below example enables PFC for code point 4 on interface 1/1/1. In times of congestion, the switch will generate a pause frame and send it to the queue that traffic marked with an 802.1p value of 4 uses.

Step	Command

Create a new schedule-profile called test SMB (no form removes)	`switch(config)# interface 1/1/2`
	
	`switch(config-if)# flow-control priority 4`

See the Aruba CX Fundamentals Guide for more details on PFC.

Configure DCBx Application TLV

The DCBx application to priority map TLV gets advertised in the DCBX application priority messages sent to attached devices. These messages tell the DCBX peer (with willing bit on) to send the application traffic with the configured priority so that the network can receive and queue traffic properly.

You can configure multiple applications in this manner. Take note if the attached device does not honor the DCBx application TLVs then the device will need to be manually configure3d to mark traffic properly.

The below example shows the DCBx Application TVL syntax, as well as some example configurations.

Step	Command
DCBx Syntax	`switch(config)# dcbx application {ISCSI | TCP-SCTP <PORT-NUM> | TCP-SCTP-UDP <PORT-NUM> | UDP <PORT-NUM> | ether <ETHERTYPE>} priority <PRIORITY>`
Example: Mapping iSCSI traffic to priority 4	`switch(config)# dcbx application iscsi priority 4`
Example: Mapping TCP Port to priority 4	`switch(config)# dcbx application tcp-sctp 860 priority 4`

See the Aruba CX Fundamentals Guide for more details about DCBx.

RoCEv2 Configuration guidance and details

RoCEv2 solutions are very similar to the RoCEv1 solution, however, the solution will now have a L3 hop in the traffic path. The hop towards the hosts will most likely still be L2, however the hop between switches will now be L3.

Because of this, that L3 link will need to trust DSCP. The receiving switch will then be able to honor the DSCP value and place that traffic in the proper queue based on the queue-profile.

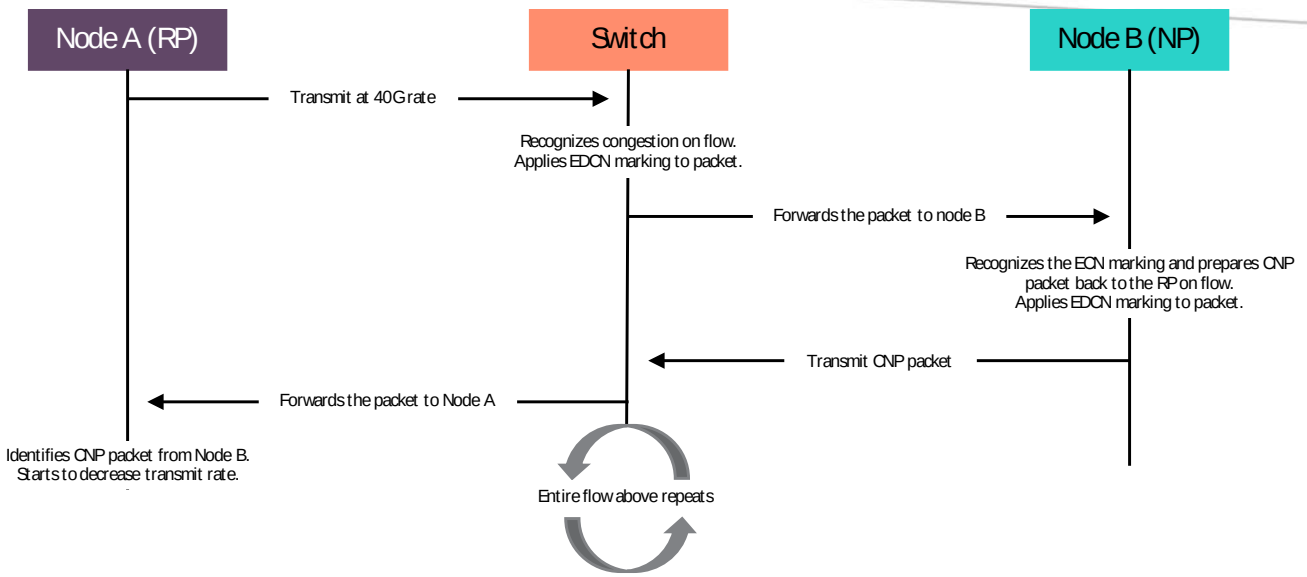
RoCE Congestion Management and ECN

RCM provides the capability to avoid congestion hot spots and optimize the throughput of the fabric even over Layer 3 links.

The RoCEv2 Congestion Management feature is composed of three points:

- The congestion point (CP): Detects congestion and marks packets using DCQN bits.
- The notification point (NP) (receiving end node): Reacts to the marked packets by sending congestion notification packets (CNPs).
- The reaction Point (RP) (transmitting end node): Reduces the transmission rate according to the received CNPs.

With RoCE RCM, when congestion occurs the CP sees congestion, but it continues to forward the traffic to the destination NP. Now that the NP destination knows there is congestion it replies to the source node RP and informs it that there is congestion. Now the source RP reacts by decreasing and later on increasing the Tx “transmit” rates according to the feedback provided. The source RP node keeps increasing the Tx rates until the system reaches a steady state of non-congested flow with traffic rates as high as possible.



The flow will continue in a loop until Node A has decreased the Tx rate to a point where:

- The switch will no longer feel congested and will stop ECN marking
- Node B will stop sending CNPs to Node A

Figure 3. RCM steps

On the networking side RoCE RCM is configured using IP ECN. Packets are marked with IP ECN bits by each switch at a configurable ECN threshold, allowing TCP to function normally and helping to prevent pauses. Both endpoints and each device in the traffic path needs to support this feature. ECN uses the DS field in the IP header to mark the congestion status along the packet transmission path.

ECN operates as follows:

- When the average queue size exceeds the lower limit, and is below the upper limit, before the device drops a packet which should be dropped according to the drop probability, the device examines the ECN field of the packet
  - If the ECN field shows that packet is sent out ECN-capable terminal, the device sets both the ECT bit and CE bit to 1 and forwards the packet
  - If the ECN field shows that a packet has experienced congestion (Both the ECT bit and CE bit are 1), the device forwards the packet without modifying the ECN field
  - If both ECT bit and CE bit are 0s, the device drops the packet
- When average queue size exceeds the upper limit, the device drops packet, regardless of whether the pack is sent from ECN-capable terminal

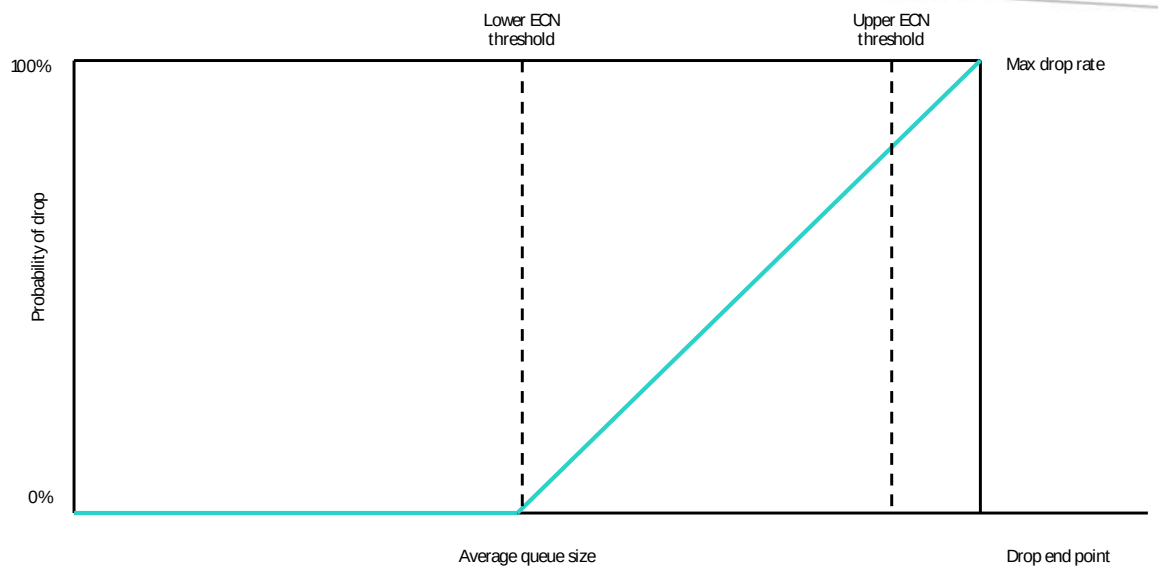


Figure 4. IP ECN thresholds

RoCEv2 Configuration

Below are the additional steps that should be taken when configuring a RoCE2 solution with Aruba CX:

1. Enable LLDP and DCBx
2. Configure QoS Queue-Profile
3. Configure Global Trust on L2 links
4. Configure QoS Schedule Profile
5. Configure PFC on interfaces that require lossless Ethernet
6. Configure Application TLV
7. Create ECN Threshold Profile and apply to proper interface/queues

Note: For ECN to work, all switches in path between two ECN-enabled endpoints must have ECN enabled

Step	Command
Create a threshold profile with ECN action on queue	config)# qos threshold-profile ECN (config-threshold)# queue 1 action ecn all threshold 50 percent (8360) (config-threshold)# queue 1 action ecn all threshold 40 kbytes (8325/8400)
(Option 1) Apply profile globally (all ports) (Option 2) Apply profile to specific	config)# apply qos threshold-profile ECN config)# int 1/1/3 (config-if)# apply qos threshold-profile ECN (config)# int lag 10

Ethernet or LAG interfaces	`(config-if)# apply qos threshold-profile ECN`
Verify threshold is applied	`(config)# show qos threshold-profile ECN`

See the Aruba CX QoS Configuration Guide for more details about ECN.

Summary

In summary, utilizing RDMA solutions such as RoCE can lead to increased performance, reduced latency, as well as lower CPU utilization.

Converged solutions, such as RoCE, carry both lossless and lossy traffic and they are gaining in popularity in modern data centers. This is thanks in large part to the growth of Hyper-convergence, IP storage and of course RoCE based solutions which have also grown thanks to solutions like SMB direct, and more.

When designing any DC, but especially a converged DC, one main point should be to keep it simple. Convergence at the ToR is the first and obvious part of a network to converge. Modern servers are now sending up massive amount of various types of traffic, some requiring lossless behavior, while other types of traffic is ok with packet drops. Upwards of 80% of the cabling in data centers is at the ToR so converging the ToR helps to reduce capital and operational expenses. This is because of fewer switches, lower power consumption, reduced cooling, and reduced NIC/HBAs and cables.

From there the simplest solution is to break off the lossless traffic from the lossy traffic at the ToR. In storage environments, this is commonly achieved by using a SAN core for the lossless traffic and the normal LAN core for lossy traffic.

However, more and more solutions, like Hyper-converged solutions may require lossless traffic to traverse the main LAN network core. The HPE Networking environments can support this, however, keep in mind that even though we can configure the network to transport lossless traffic in an efficient manner we cannot forget to also provision the network for peak loads.

As discussed, protocols like PFC can create a cascading pausing affect which can affect unrelated traffic. Ensuring the network links are provisioned for peak load times will help to reduce excessive congestion.

Ensuring traffic is segregated into proper queues can also help. As an example, perhaps the environment has chatty servers, initiators, and targets. This might mean moving chatty servers, initiators and targets into a different queue so that pauses in that queue will not affect other traffic.

Table 1. Summary.

Feature	RoCEv1	RoCEv2	Notes
Priority-based Flow Control	YES	YES	Must use always. If not used, in times of congestion the RDMA advantages will not be achieved.
Enhanced Transmission Selection (ETS)	YES	YES	Must use in converged environments. If not used, lossless traffic classes can get starved for bandwidth.
Data Center Bridging Exchange (DCBX)	YES	YES	Not Mandatory but recommended.
Quantized Congestion Notification (QCN)	YES	NO	Highly recommended for L2 RoCEv1. CN helps to address pause unfairness and victim flow issues. When used with PFC, PFC acts as fast acting mechanism to address microbursts, while CN smooths out traffic flows helping to reduce pause storms under normal load.

IP Explicit Congestion Notification (ECN)	NO	YES	Highly recommended for L2 RoCEv2. ECN will help to address pause unfairness and victim flow issues. When used with PFC, PFC acts as fast acting mechanism to address microbursts, while CN smooths out traffic flows helping to reduce pause storms under normal load.

Resources, contacts, or additional links

3333 Scott Blvd. | Santa Clara, CA 95054
1.844.472.2782 | T: 1.408.227.4500 | FAX: 1.408.227.4550 | info@arubanetworks.com