Open. Together.
Disaggregation & Data Center TCP: Maximizing OCP Datacenter Efficiency Through the Reduction of Tail Latency

Nicolaas Viljoen, Associate Director Software Engineering, Netronome
Amal Tariq, Network Hardware Engineer, Facebook
Lawrence Brakmo, Kernel Software Engineer, Facebook
Motivation for Disaggregation

**Flash Disaggregation:** moves memory access from within a server to the network

**Stranded Capacity:** Disaggregated flash storage improves utilization by up to 40% [1]

The Yosemite v2 & Disaggregation

**Power efficiency:** The Yosemite v2 increases compute per watt of standard designs.

**To combine with disaggregation:** Must handle bursty traffic while minimizing drops to reduce tail latency. This is mainly due to in-cast related network stress.

---

**DCTCP + ECN**
- ECN (explicit congestion notification): aware of extent (rather than just presence) of congestion.
- Allows sender to respond to congestion before packets are dropped.

---

[Diagram showing network components: NIC, switch(es), 12-24G PCIe (each link), 50G Ethernet, database.]
DCTCP/ECN in Disaggregated Architecture

2. When buffer used reaches ECN marking threshold, NIC adds ECN flag to packet

5. Buffering handles packets in flight, minimizing drops

4. ECN flag reduces the congestion window

6. Window size is maintained at usable level without packet loss

3. Returning ACK marked with ECN flag

1. Storage head node opens up increasing window

Case Studies
Throughput with DCTCP

Often a concern raised with DCTCP is that available bandwidth is underutilized. This is due to the low ECN threshold values used.

However, with sufficient buffer space to set higher thresholds, link underutilization is not an issue.

This removes the throughput regressions otherwise observed.
Initial tests with DCTCP: setup

- **Wedge 100S switch**
  - 50G
  - 25G
  - Tioga Pass headnode
  - Lightning storage
  - Agilio-CX NIC
  - Leopard compute node

- **Wedge 100S switch**
  - 50G
  - 25G
  - Tioga Pass headnode
  - IPerf Server
  - Agilio-CX NIC
  - Leopard compute node
  - 12G (PCIe x2)
Initial tests with DCTCP

Tested in production environment, shadowing latency-sensitive application traffic.

**DCTCP traffic** is the storage reads between compute and head node *(traffic under our control)*. **CUBIC traffic** is everything else (e.g. application queries to compute node or external storage writes) *(traffic not under our control)*.

Latency as primary performance metric, other metrics were not to regress (throughput, QPS, etc.)

PCIe at the NIC was limited to **mimic in-cast** problem.
Split Queues by Congestion Control

Latency peaks correspond to daily partition (CUBIC traffic)

Due to CUBIC traffic dominating, buffer queues are not able to respond to ECN

Resolved this by creating separate buffer queues for DCTCP and CUBIC traffic in the NIC

Both sets of queues were tuned to ensure fairness
Split Queues by Congestion Control

- Headnode receive traffic
- DCTCP host, **shared queue**
- DCTCP host, **split queues**
Multi-Host NIC tests: architecture

Yosemite v2

- Wedge 100S switch
- Yosemite v2 compute node + Netronome Agilio-CX 50G OCP NIC
- Tioga Pass head node
- Lightning storage

Yosemite v2

Agilio-CX 50G NIC

PCIe x4 Twin Lakes

Logic could be replaced by offloaded BPF

50G link

L2 Switch

classifier

PCIe x4 Twin Lakes

PCIe x4 Twin Lakes

PCIe x4 Twin Lakes
Multi-Host NIC tests

Wedge 100S switch

4x 50G

Tioga Pass headnode

Lightning storage

Agilio-CX NIC

Yosemite v2 compute node

24G (PCIe x4)

4x

Single compute slot in Yosemite active

All 4 compute slots active
Multi-Host NIC tests: added stress

- **Wedge 100S switch**
  - 4x 50G
  - 4x 50G
  - 50G

- **Tioga Pass headnode**
  - Lightning storage

- **IPerf Server**

- **Agilio-CX NIC**
  - 24G (PCIe x4)

- **Yosemite v2 compute node**

*Single slot active with Iperf traffic (pulsed on/off hourly)*

*All 4 slots active with Iperf traffic (pulsed on/off hourly)*
ECN in Switch

High latencies due to **congestion in switch**.

Need a solution for congested switches in conjunction with NICs.

Had previously tested **enabling ECN in switches** for use with DCTCP, these tests confirmed the advantages of enabling ECN and using DCTCP across rack.

We will go over these tests in the following slides.
DCTCP in Switch: Topology

6 rack tests
- 3 racks are store servers
- 3 racks (workers) read data from store servers
- Cross traffic between workers
# DCTCP in Switch: Benchmarks

<table>
<thead>
<tr>
<th></th>
<th>CUBIC</th>
<th>DCTCP</th>
</tr>
</thead>
<tbody>
<tr>
<td>FSW to Worker Avg Link Util %</td>
<td>69.9</td>
<td>69.8</td>
</tr>
<tr>
<td>Storage CPU (%)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Worker CPU (%)</td>
<td>Y</td>
<td>Y+1%</td>
</tr>
<tr>
<td>FSW Discards (bits)</td>
<td>89M</td>
<td>235K (0.3%)</td>
</tr>
<tr>
<td>Worker rack discards (bits)</td>
<td>417M</td>
<td>0</td>
</tr>
<tr>
<td>Storage Retransmits</td>
<td>0.020 %</td>
<td>0.000 %</td>
</tr>
<tr>
<td>Worker Retransmits</td>
<td>0.173 %</td>
<td>0.078 %</td>
</tr>
<tr>
<td>Storage ECN CE Marked (%)</td>
<td>6.5</td>
<td></td>
</tr>
<tr>
<td>Worker ECN CE Marked (%)</td>
<td>12.8</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** If we increase load until link utilization is 99%:
- FSW discards in CUBIC are 160B vs. 157M (0.1%) under DCTCP
- Storage retransmits are .6% under CUBIC vs. 0.001% under DCTCP
Multi-Host NIC + ECN in Switch

Suspected switch bottlenecks because all hosts used (including iperf servers) were in the same rack.

Enabled ECN marking in the switch and were able to significantly reduce tail latency.
Summary

Linking advances in congestion control with OCP based SmartNICs reduces tail latency significantly.

This allows OCP Yosemite v2 systems to be used in a wider variety of use cases, significantly improving efficiency.

Without also implementing ECN/DCTCP in the switches it is possible to construct cases with high latency.

Combining ECN/DCTCP in the multihost NICs and in RSWs, it is possible to ‘guarantee’ low tail latency.
Product/Facility Info

Wedge100S
https://www.opencompute.org/wiki/Networking/SpecsAndDesigns#Facebook_Wedge_100S_32x100G
https://www.opencompute.org/products/190/edgecore-networks-wedge100s-100gbe-data-center-switch

Yosemite v2
https://www.opencompute.org/products/275/wiwynn-yosemite-v2

Agilio-CX 50G OCP NIC
https://www.opencompute.org/wiki/Server/Mezz#Specifications_and_Designs
Call to Action

Netdev (kernel): netdev@vger.kernel.org
Mezz: opencompute-mezz-card@lists.opencompute.org
Server: opencompute-server@lists.opencompute.org
Switch: opencompute-networking@lists.opencompute.org

Additional Information: