Ethernet Accelerated Machine Learning Pods

Accelerating your GPU Pods

David Iles | SC 2019
RoCE Accelerated AI Solutions

RDMA Supercharges Leading AI Frameworks

| 60% Higher ROI | 2.5X Better Performance | Up to 95% Scaling Efficiency | Up to 50% Savings on Capital & Operation Cost |

---

© 2019 Mellanox Technologies
GPUDirect™ RoCE Technology

With GPUDirect

No GPUDirect

GPUDirect™ Powered by RoCE

Delivers 10X Better Performance

© 2019 Mellanox Technologies
What Makes a Great Ethernet AI Switch?

Simple Configuration
- 1 Command CLI config
- 1 Click GUI config

High Performance
- High PPS & Low latency
- Fair & Predictable performance

Advanced Congestion Control
- Early detection and prevention
- RoCE over VXLAN

Extensive Visibility
- Single pane-of-glass
- Real time RoCE Telemetry
What Makes a Great Ethernet AI Switch?

Simple Configuration
- 1 Command CLI config
- 1 Click GUI config

High Performance
- High PPS & Low latency
- Fair & Predictable performance

Advanced Congestion Control
- Early detection and prevention
- RoCE over VXLAN

Extensive Visibility
- Single pane-of-glass
- Real time RoCE Telemetry
1 Click RoCE Provisioning

- Automated RoCE config across entire fabric
  - No manual configuration needed

- End-to-End configuration
  - Mellanox switches
  - Mellanox NICs

RoCE End to End

Network
- Lossless
- Resilient

Force Configuration on Existing LAGs

Supported only on Spectrum Switches and Linux Hosts.

Close  Submit
RoCE Made Easy

Mellanox “Do RoCE”

switch (config) # roce

Other’s RoCE Configuration

Step 1 – Ingress Traffic Classification
class-map type qos match-all CNP
match dscp 48
class-map type qos match-all RDMA
match dscp 26
policy-map type qos QOS_MARKING
class RDMA
set qos-group 3
class CNP
set qos-group 6

Step 2 – Configure QoS Policies
policy-map type network-qos QOS_NETWORK
class type network-qos c-8q-nq
pause pfc-qos 3
mtu 2240
policy-map type queuing QOS_QUEUEING
class type queuing c-out-8q-q3
random-detect minimum-threshold 150
kbytes maximum-threshold 1500
kbytes drop-probability 100 weight 0 ecm
bandwidth remaining percent 20
class type queuing c-out-8q-q6
priority level 1
policy-map type queuing INPUT_QOS_QUEUEING
class type queuing c-in-q3
queue-limit dynamic 3
system qos
service-policy type queuing input INPUT_QOS_QUEUEING
service-policy type queuing output QOS_QUEUEING
service-policy type network-qos QOS_NETWORK

Step 3 – Configure Resource Allocation
hardware access-list tcam region e-racl 0
hardware access-list tcam region vpc-convergence 0
hardware access-list tcam region racl-lite 768
hardware access-list tcam region l3qos-intra-lite 0
hardware access-list tcam region qos 256
hardware access-list tcam region e-qos 256

Step 4 – Set Per Port Configuration
interface Ethernet1/1-1/32
Service-policy type qos input QOS_MARKING

24 Lines!
What Makes a Great Ethernet AI Switch?

Simple Configuration
- 1 Command CLI config
- 1 Click GUI config

High Performance
- High PPS & Low latency
- Fair & Predictable performance

Advanced Congestion Control
- Early detection and prevention
- RoCE over VXLAN

Extensive Visibility
- Single pane-of-glass
- Real time RoCE Telemetry
Fully Shared Buffers are Superior for RoCE

**Fully Shared Packet Buffer**

- Microburst occurs here with 16MB of packet buffers to absorb burst

**Competitor’s multi-core based buffer scheme**

- Microburst occurs here, but only 25% of buffers are available to any one port

Superior Micro Burst Performance
Spectrum’s Fully Shared Buffer Provides **4X effective buffer size**!
Superior Microburst Absorption

![Graph showing Microburst Absorption comparing Spectrum-2, Trident-3, and Tomahawk-2]

- **Microburst Absorption**
  - Packet Sizes: 256B, 512B, 1518B
  - MEGABYTES: 0.00 to 35.00

Fully Shared Packet Buffers

- Spectrum-2 vs Trident-3 vs Tomahawk-2

© 2019 Mellanox Technologies
Latency at 100G

RFC2544 Average Latency Results 100G-100G (Lower is better)

Spectrum-2 vs Tomahawk-2 vs Trident-3
Latency at 25G

bandwidth < 99%, TD3 800-920ns latency grows per packet size
### What Makes a Great Ethernet AI Switch?

<table>
<thead>
<tr>
<th>Feature</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Simple Configuration</strong></td>
<td>• 1 Command CLI config</td>
</tr>
<tr>
<td></td>
<td>• 1 Click GUI config</td>
</tr>
<tr>
<td><strong>High Performance</strong></td>
<td>• High PPS &amp; Low latency</td>
</tr>
<tr>
<td></td>
<td>• Fair &amp; Predictable performance</td>
</tr>
<tr>
<td><strong>Advanced Congestion Control</strong></td>
<td>• Early detection and prevention</td>
</tr>
<tr>
<td></td>
<td>• RoCE over VXLAN</td>
</tr>
<tr>
<td><strong>Extensive Visibility</strong></td>
<td>• Single pane-of-glass</td>
</tr>
<tr>
<td></td>
<td>• Real time RoCE Telemetry</td>
</tr>
</tbody>
</table>
Scaling RoCE with ECN

Explicit Congestion Control
- ECN = Flow Level Congestion Handling
- ECN throttles A+B+C+D
- Victim traffic from X passes
Increasing RoCE Performance

Faster Congestion Recovery

- **Fast Congestion Notification**
  - Packets marked as they leave queue
  - Faster congestion recovery
  - Reduces average queue depth
    - Lowers real world latency
    - *Improves application performance*

- **Legacy Congestion Notification:**
  - Packets marked as they enter queue
  - Notification delayed until queue empties
  - Up to 5msec slower alerts

**Diagram:****
- Marks packets entering queue
- Marks packets exiting queue
RoCE + VXL AN On the Same Network

RoCE with VXL AN Fabric

- ROCE + VXL AN
  - Quality of Service within VXL AN tunnels
  - Preserve ECN even with VXL AN
  - Unique Mellanox differentiator
  - Other vendors don’t copy QOS between outer & Inner headers

- Use Cases: VXL AN +
  - GPU PODs
  - Machine Learning
  - NVMe over Fabrics
  - Storage Spaces Direct
  - Spark
What Makes a Great Ethernet AI Switch?

Simple Configuration
- 1 Command CLI config
- 1 Click GUI config

High Performance
- High PPS & Low latency
- Fair & Predictable performance

Advanced Congestion Control
- Early detection and prevention
- RoCE over VXLAN

Extensive Visibility
- Single pane-of-glass
- Real time RoCE Telemetry
Easy RoCE Diagnostics

Extensive Visibility
Provides real-time RoCE configuration and stats

Proactive Monitoring
Identifies network issues as they occur, and generate alerts to the user

Fast Troubleshooting
Enables the user to quickly find root cause and resolve the issue
Single Pane-of-Glass for RoCE

RoCE Config Made Easy
Single Pane RoCE Monitoring
Fast Troubleshooting
Built for Integration

RoCE Network Fabric
Easy RoCE Diagnostics

Show RoCE Interface:
- RoCE Traffic
  - Packet & Byte Count
  - Packet drops
  - Current & max buffer usage
- Congestion Notification Packets:
  - Packet & Byte Count
  - Packet drops
  - Current & max buffer usage
- ECN:
  - ECN marked packets

Show RoCE Global:
- RoCE Configuration
  - RoCE Mode (Lossless/lossy)
  - Buffer settings
  - Port congestion control settings
  - PFC settings
- Priorities to Buffers Mapping
  - TC mapping
  - Which Queues are for ROCE
  - ETS settings
- Buffer Pools
  - Buffer memory allocation
  - Max buffer usage
  - Type: Lossy/lossless
### Show RoCE Interface

```
switch (config) # show interfaces ethernet 1/1 counters roce
```

<table>
<thead>
<tr>
<th>Rx:</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>RoCE PG packets</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE PG bytes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE no buffer discard</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP PG packets</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP PG bytes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP no buffer discard</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE PFC pause packets</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE PFC pause duration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE buffer usage (bytes)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE buffer max usage (bytes)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP buffer usage (bytes)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP buffer max usage (bytes)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Tx:</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>ECN marked packets</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE TC packets</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE TC bytes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE unicast no buffer discard</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP TC packets</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP TC bytes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP unicast no buffer discard</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE PFC pause packets</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE PFC pause duration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE buffer usage (bytes)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>RoCE buffer max usage (bytes)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP buffer usage (bytes)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>CNP buffer max usage (bytes)</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Why Do We Need Telemetry?

- Faster Time to Innocence
- Faster Time To Resolution
- Get more out of the Network
WJH™ Accelerates the Time to Root-Cause
WJH – How Does It Work?

1. SDK generates:
   WJH messages

2. WJH Agent:
   Streams to a Database

3. Presentation layer shows:
   What Just Happened

The Important Questions

WHO is being impacted
WHAT is causing the problem
WHEN it happened
WHERE is the problem
WHY it is happening
Root Cause + how to fix it
What Do We Monitor?

**L1**
- Port down reason
- Flaky cable

**L2/L3/Overlay**
- Blackhole
- VLAN Mismatch

**Buffer**
- Tail drop

**ACLs**
- Deny based on IP
- Deny based on VLAN

**Congestion**
- Incast
- Busy storage device

**Latency**
- Crossing threshold
- Pause frames

**RoCE**
- Wrong priority identification
- PFC watchdog

**Load Balance Validation**
- Suboptimal ECMP
- Suboptimal LAG
RoCE Monitoring with Mellanox WJH™

Flow Visualization

Congestion Monitoring

Latency Monitoring

Fault Detection

Performance Monitoring

Root Cause Analysis

© 2019 Mellanox Technologies
ScaleMatrix: ONTAP AI Deployed Anywhere

NetApp partners with ScaleMatrix to deliver ONTAP AI as a plug-and-play solution that can be deployed anywhere
Combining edge capabilities and mobility/modularity of ScaleMatrix and DDC Cabinet Technology with the power of ONTAP AI

Challenge:
- Edge locations and many office environments lack datacenter capability

Solution:
- **ONTAP AI** delivered in mobile (R-Series) and modular (S-Series) DDC Cabinet Technology from ScaleMatrix
- Provides a self-contained environment, with guaranteed air flow, integrated security, fire and noise suppression

Key use cases:
- Edge inferencing in retail, healthcare, manufacturing
- AI and other high-performance workloads

Customer benefits:
- Can be deployed in any environment – no datacenter required
- Up and running within minutes – plug in and power on
- Ability to redeploy elsewhere, with ease

ScaleMatrix at SC19, booth #2131 - showcasing 3 solutions based on ONTAP AI
Open Ethernet SN2000 Series

- **SN2700 – 32x100G**
  The Ideal 100G ToR / Aggregation

- **SN2410 – 48x25G + 8x100G**
  25G ➔ 100G ToR

- **SN2100 – 16x100G ports**
  (up to 32x50G, 64x25/10G)
  Ideal storage / Database 25/100G Switch

- **SN2010 – 18x25G + 4x100G**
  Ideal Hyperconverged Switch
  10/25G ➔ 100G half width ToR

- Predictable Performance
- Fair Traffic Distribution for Cloud
- Best-in-Class Throughput, Latency, Power Consumption
- Zero Packet Loss

**300ns**
SN2700 – 169W
SN2410 – 165W
SN2100 – 94W
SN2010 – 57W

Energy efficiency
SN3000 Series Switch Family

**SN3800 – 64x100G**
Scalable 100G Spine Switch

**SN3700C – 32x100G**
Upgraded 100G Spine and high-density ToR Switch

**SN3420 – 48x25G + 12x100G**
25G ➔ 100G Non-Blocking ToR Switch

**SN3700 – 32x200G**
50G/200G PAM4 Spine Switch

**SN3510 – 48x50G + 6x400G**
50G ➔ 200G/400G PAM4 ToR Switch
Thank You