A Lossless Network for Data Acquisition
2016 Real Time Conference

07.06.2016
Grzegorz Jereczek
Outline

1. Introduction
2. A lossless switch for data acquisition networks
3. A lossless network for data acquisition
4. Conclusions and outlook
Introduction
Incast congestion in data acquisition networks

Synchronized many-to-one bursts from ROS overflow packet buffers in the network.

![Graph showing event rate vs. number of event fragments requested]

- **Event rate (Hz)**
- **No. of event fragments requested**

**Graph Details:**
- **Throughput collapse**
- **Requested rate**
- **ATLAS TDAQ**

**Diagram Components:**
- **ROS Readout system**
- **DAQ network**
- **Data collector DCM**

**Notes:**
- Grzegorz Jereczek
- ICE-DIP Project
- 07.06.2016
General approaches

**Flow control**: Ethernet Pause/PFC, InfiniBand
Designed to absorb fluctuations, HoL blocking

**Congestion control**: traffic shaping, TCP variants, Ethernet DCB
HW/SW support, dependent on fragment sizes/counts and network architecture, sender-side buffering

**Deep buffers**
Best throughput, simple push architecture, but rare and/or expensive devices
General approaches

Flow control: Ethernet Pause/PFC, InfiniBand
Designed to absorb fluctuations,
HoL blocking

Congestion control: traffic shaping, TCP variants, Ethernet DCB
HW/SW support, dependent on fragment sizes/counts and
network architecture, sender-side buffering

Deep buffers
Best throughput, simple push architecture,
but rare and/or expensive devices

Can we use the DRAM memory as a packet buffer?
COTS-servers as network switches for DAQ

High I/O performance of modern servers

Memory: 540 Gbps  (DDR4-2133, 4 channels/CPU)
PCIe: 63 Gbps, even 10 slots on a board  (PCIe Gen3 x8)
COTS-servers as network switches for DAQ

✓ High I/O performance of modern servers

Software availability
   Production quality software switch: Open vSwitch (OvS)

Frameworks for fast packet processing: DPDK

Network control: Software Defined Networking (SDN)
COTS-servers as network switches for DAQ

✓ High I/O performance of modern servers

✓ Software availability

Production quality software switch: Open vSwitch (OvS)
→ Optimize for throughput

Frameworks for fast packet processing: DPDK
→ Buffering mechanism

Network control: Software Defined Networking (SDN)
→ Use the global view of the network

Goal: Lossless network based on software switches with large packet buffers in DRAM optimized for DAQ
A lossless switch for data acquisition networks
Optimizing Open vSwitch for DAQ

Some optimizations to datapath for high-throughput.

Queueing

Packets queued in the DPDK’s rings.

A single ring dedicated to a single DCM.

Rings are distinct ports *(daqring port)*.
Optimizing Open vSwitch for DAQ

Some optimizations to datapath for high-throughput.

Queueing

Packets queued in the DPDK’s rings.

→ Size of an event

A single ring dedicated to a single DCM.

Rings are distinct ports (*daqrings port*).
Optimizing Open vSwitch for DAQ

Some optimizations to datapath for high-throughput.

Queueing

Packets queued in the DPDK’s rings.
→ Size of an event

A single ring dedicated to a single DCM.
→ Hundreds of rings for the entire system
→ Rate limitation possible

Rings are distinct ports (daqring port).
Optimizing Open vSwitch for DAQ

Some optimizations to datapath for high-throughput.

Queueing

Packets queued in the DPDK’s rings.
→ Size of an event

A single ring dedicated to a single DCM.
→ Hundreds of rings for the entire system
→ Rate limitation possible

Rings are distinct ports (daqring port).
→ Programming and optimizing flows with OVSDB and OpenFlow
12 x 10GbE prototype

Note: all results for large packets (MTU: 1500B).
All-to-all incast: 12 ROSes and 144 DCMs

No packet drops: lossless operation.

98% of theoretical goodput with 8 CPU cores.

Utilizing full bidirectional bandwidth of 120Gbps.

Bandwidth-wise, comparable to ATLAS DAQ in run 1.

Goodput = \[\frac{\text{event data collected}}{\text{collection time}}\]

Grzegorz Jereczek
ICE-DIP Project
07.06.2016
The lossless software switch outperforms regular switches with hardware flow control

OvS with DPDK daqrings, no CC
The lossless software switch outperforms regular switches with hardware flow control

- OvS with DPDK daqrings, no CC
- Vendor A, TCP cubic
- Vendor B, TCP cubic

Grzegorz Jereczek
ICE-DIP Project
07.06.2016
The lossless software switch outperforms regular switches with hardware flow control.

Grzegorz Jereczek
ICE-DIP Project
07.06.2016
A lossless network for data acquisition
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:
Data flow from ROS (R) to racks of DCMs (H).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:
Data flow from ROS (R) to racks of DCMs (H).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:
Data flow from ROS (R) to racks of DCMs (H).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:
Data flow from ROS (R) to racks of DCMs (H).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:
Data flow from ROS (R) to racks of DCMs (H).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

**Applying in DAQ:**

Data flow from ROS (R) to racks of DCMs (H).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:

Data flow from ROS (R) to racks of DCMs (H).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:
Data flow from ROS (R) to racks of DCMs (H).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:
Data flow from ROS (R) to racks of DCMs (H).

OpenFlow L3-only network.
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:

Data flow from ROS (R) to racks of DCMs (H).

OpenFlow L3-only network.

DCM flows distributed across available paths and daqrings (waterfilling).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:
Data flow from ROS (R) to racks of DCMs (H).

OpenFlow L3-only network.

DCM flows distributed across available paths and daqrings (*waterfilling*).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:

Data flow from ROS (R) to racks of DCMs (H).

OpenFlow L3-only network.

DCM flows distributed across available paths and daqrings (*waterfilling*).
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:

Data flow from ROS (R) to racks of DCMs (H).

OpenFlow L3-only network.

DCM flows distributed across available paths and daqrings (*waterfilling*).

OvS also on the end-nodes.
Parallel leaf-spine planes

Topology based on Facebook’s datacenter fabric.

Applying in DAQ:

Data flow from ROS (R) to racks of DCMs (H).

OpenFlow L3-only network.

DCM flows distributed across available paths and daqrings (*waterfilling*).

OvS also on the end-nodes.

No need to use ECMP, LAG, or MLAG (no hashes!).
An example: offered DAQ bandwidth

Pods: 4
ROSes in a pod: 25
DCM racks in a pod: 7

Planes: 1
Planes: 2
Planes: 3
Planes: 4

DAQ bandwidth [Tbps]

No. of spine switches in a plane
Prototype topology (8 switches)
Offered DAQ bandwidth (theory)

Pods: 2
ROSes in a pod: 3
DCM racks in a pod: 3

No. of spine switches in a plane
0
50
100
150
200
250

DAQ bandwidth [Gbps]
Planes: 1
Planes: 2
Planes: 3
Planes: 4

No. of spine switches in a plane
Offered DAQ bandwidth (theory)

Pods: 2
ROSes in a pod: 3
DCM racks in a pod: 3

No. of spine switches in a plane
0 50 100 150 200 250
DAQ bandwidth [Gbps]
Planes: 1
Planes: 2
Planes: 3
Planes: 4

40 60 120

No. of spine switches in a plane
Offered DAQ goodput (actual)

- **Theory**
- **Actual**

<table>
<thead>
<tr>
<th>No. of spine switches in a plane</th>
<th>DAQ goodput [Gbps]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>38</td>
</tr>
<tr>
<td>40</td>
<td>57</td>
</tr>
<tr>
<td>80</td>
<td>114</td>
</tr>
<tr>
<td>120</td>
<td>17 17</td>
</tr>
<tr>
<td>17 17</td>
<td>17 26</td>
</tr>
</tbody>
</table>

Grzegorz Jereczek
ICE-DIP Project
07.06.2016
A problem: PCIe gen1 in the end nodes

Solution:

- Rate-limited daqrings

10% 50% 90%

Percentile

1 10 100 1000

Latency [ms]

1 plane, 1 spine
1 plane, 2 spines
2 planes, 2 spines
A problem: PCIe gen1 in the end nodes

Solution: Rate-limited daqrings

- 1 plane, 1 spine
- 1 plane, 2 spines
- 2 planes, 2 spines

Percentile

Latency [ms]

1 plane, 1 spine
1 plane, 2 spines
2 planes, 2 spines
Offered DAQ goodput (actual)

With rate-limited daqrings performance improved, but still limited (see 2 planes). Limit set by PCIe gen1.

![Graph showing DAQ goodput vs. number of spine switches in a plane]
Offered DAQ goodput (actual)

With *rate-limited daqrings* performance improved, but still limited (see 2 planes). Limit set by PCIe gen1.

Default TCP congestion control (*TCP Cubic*) performs poorly.
Conclusions and outlook
Trying to prevent incast congestion in DAQ

DRAM memory provides large enough and cheap packet buffers.

Dedicated queueing to optimize the entire network.

First prototype offers **lossless operation** and **120Gbps bandwidth** for DAQ-specific network traffic with a single server.

Second prototype demonstrates the configuration and management of a **larger topology**.
Outlook

Generalized algorithm for load balancing.

Different service disciplines of DCM queues.

Fault tolerance.

Achievable port density.
Questions?
Backup
The lossless software switch outperforms regular switches with hardware flow control

OvS with DPDK daqrings, no CC
The lossless software switch outperforms regular switches with hardware flow control

- OvS with DPDK daqrings, no CC
- Vendor A, TCP cubic
- Vendor B, TCP cubic
The lossless software switch outperforms regular switches with hardware flow control.
The lossless software switch outperforms regular switches with hardware flow control.

- OvS with DPDK daqrings, no CC
- Vendor A, TCP cubic
- Vendor A, Pause, no CC
- Vendor B, TCP cubic
- Vendor B, Pause, no CC

Requested load: 70%
The lossless software switch outperforms regular switches with hardware flow control

- OvS with DPDK daqrings, no CC
- Vendor A, TCP cubic
- Vendor A, Pause, no CC
- Vendor B, TCP cubic
- Vendor B, Pause, no CC

Requested load: 99%

Grzegorz Jereczek
ICE-DIP Project
07.06.2016 17
Power consumption

Min. 95% of theoretical DAQ goodput in all cases.

Can be further optimized (less polling).

---

**Av. power per port [W]**

- **ToR A**
  - CPU freq. 2.7 GHz
  - CPU freq. 2.0 GHz
  - CPU freq. 1.2 GHz

- **ToR B**

---

**No. of CPU cores**

8 12

---

Grzegorz Jereczek
ICE-DIP Project
07.06.2016
Performance penalty with daqrings?

Better fairness among all data collectors.

More CPU cycles required due to additional port send/recv and OpenFlow lookups.
Performance penalty with *daqrings*?

Better fairness among all data collectors.

More CPU cycles required due to additional port send/recv and OpenFlow lookups

![Graph showing latency comparison between OvS with DPDK daqrings and OvS without DPDK daqrings.](image)

- **OvS with DPDK daqrings**
- **OvS without DPDK daqrings**
Optimizing for a lossless network
Rate-limited daqring

Polling interval [us]

DAQ goodput [Gbps]

max. daqring rx burst = 1
max. daqring rx burst = 2
max. daqring rx burst = 4

Grzegorz Jereczek
ICE-DIP Project
07.06.2016
The future

New family of Intel Ethernet products:

**FM10000**

Provides multiple Ethernet ports **AND** host PCIe interfaces.

**Example - FM10840:**

- 36 x 10GbE ports,
- 4 x 8-lane PCIe gen3 interfaces,
- Approx. 200 Gbps over PCIe,

*Ethernet Multi-host Controllers*

Perfect match for building larger topologies with packet buffers in host memory?