The bridge to possible

Fairness through a picosecond lens

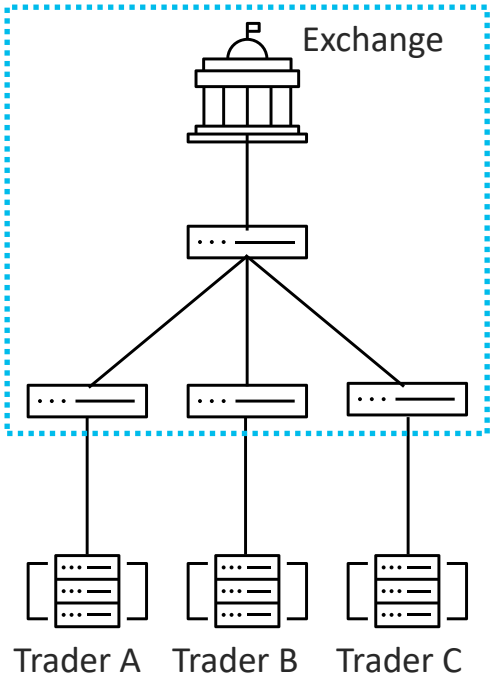
Daniel Brown
Technical Solutions Architect
STAC Summit – May 19, 2022

Agenda

- Problem – Market Data Distribution
- Why does this happen in the ASIC?
- Can FPGA be of help?
- How was the delay measured?

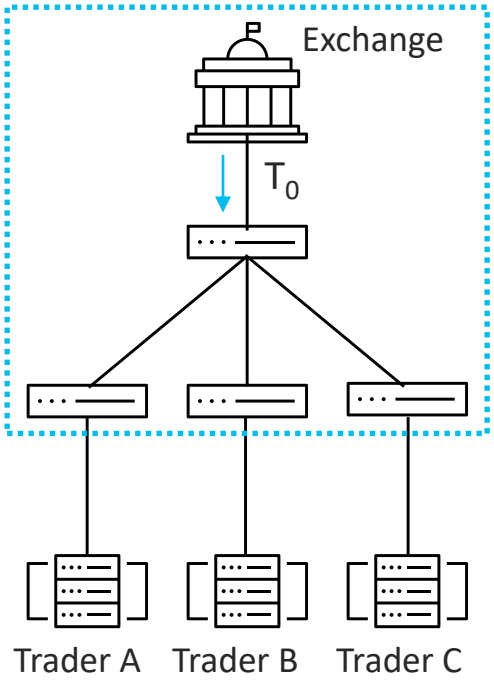
Problem – Market Data Distribution

Problem – Market Data Distribution



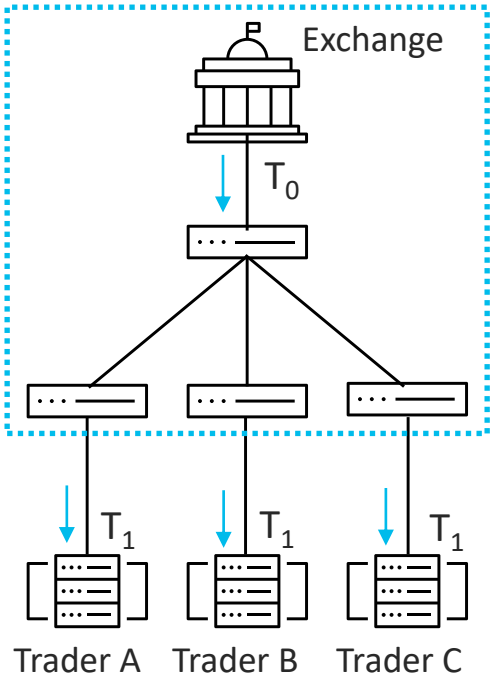
- Exchange provides market data to each trader

Problem – Market Data Distribution



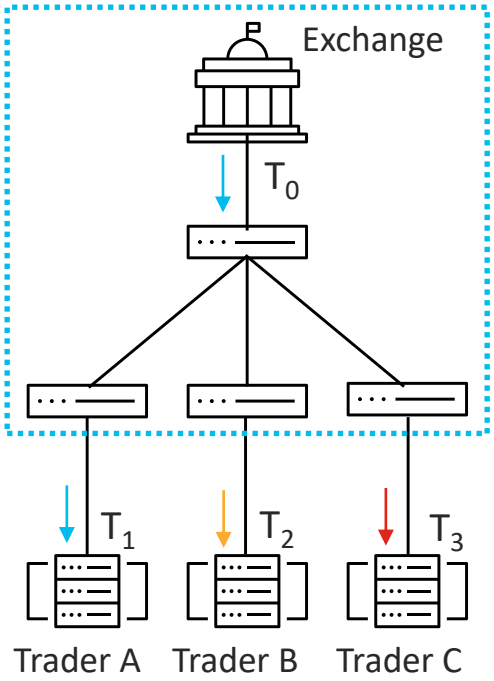
- Exchange provides market data to each trader
- Exchange distributes data at time T_0

Problem – Market Data Distribution



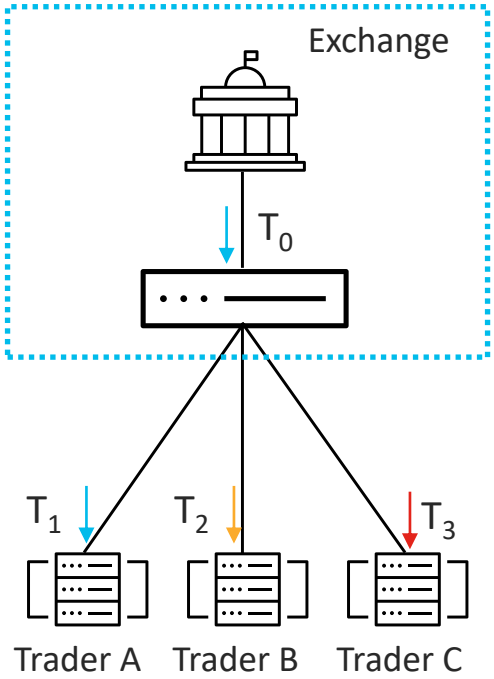
- Exchange provides market data to each trader
- Exchange distributes data at time T_0
- Assumption is that each trader receives market data at same time, time T_1

Problem – Market Data Distribution



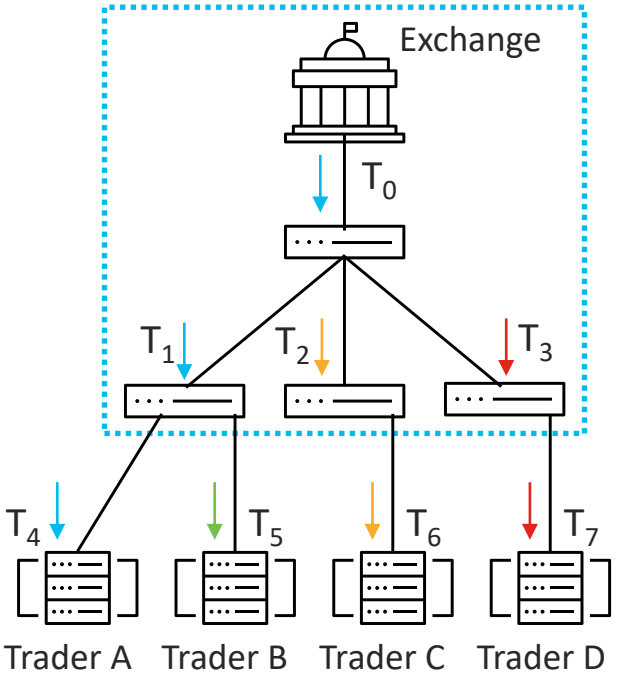
- Exchange provides market data to each trader
- Exchange distributes data at time T_0
- Assumption is that each trader receives market data at same time, time T_1
- However, these traders could receive data at different times

Network Node Delay



- In ASIC based switches, delay is product of multicast traffic forwarding
- Replication of packets is done serially to the ports
- Order and delay are product of ASIC architecture
- This will lead to delay between ports

Problem – Market Data Distribution

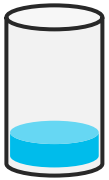


- Unfairness happens because of the network and switch architecture
- Each network node, can introduce small delay in the network path
- With multiple network hops traders may receive delayed market data

Why is this happening in the ASIC?

Why is this happening in the ASIC?

Multicast Buffer



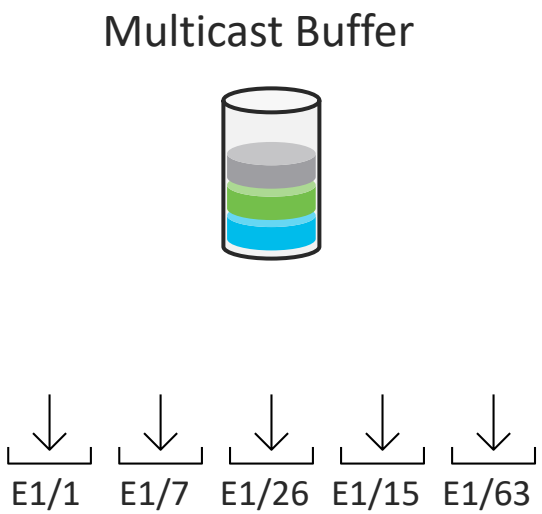
- Packets are stored in the multicast buffer in the ASIC

Why is this happening in the ASIC?

Multicast Buffer

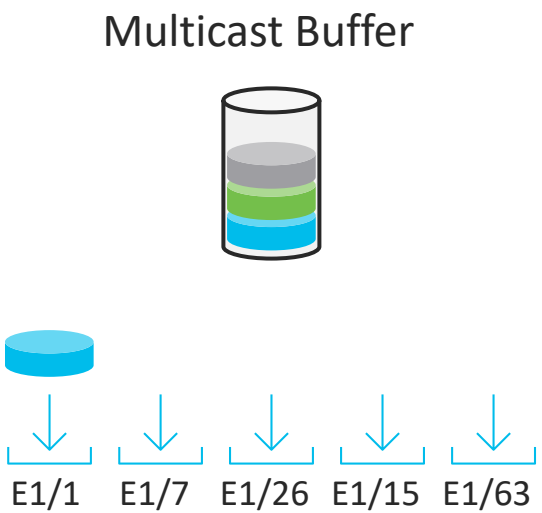
- Packets are stored in the multicast buffer in the ASIC
- If multiple packets are processed, all stored in the same buffer

Why is this happening in the ASIC?



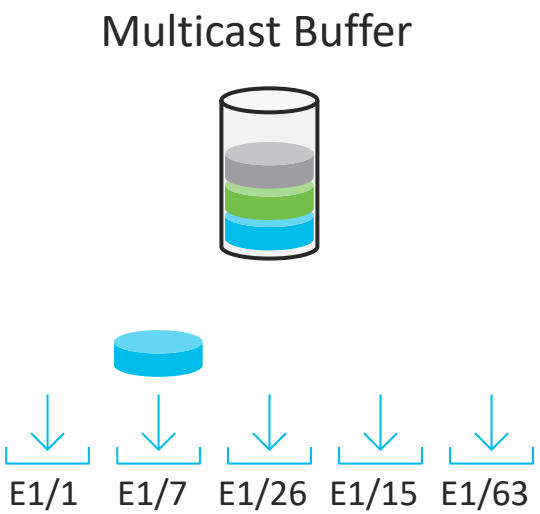
- Packets are stored in the multicast buffer in the ASIC
- If multiple packets are processed, all stored in the same buffer
- Packets are replicated, by reading packets from multicast buffer

Why is this happening in the ASIC?



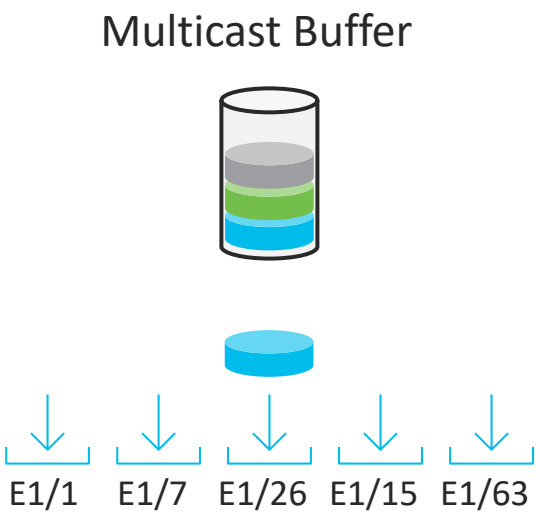
- Packets are stored in the multicast buffer in the ASIC
- If multiple packets are processed, all stored in the same buffer
- Packets are replicated, by reading packets from multicast buffer

Why is this happening in the ASIC?



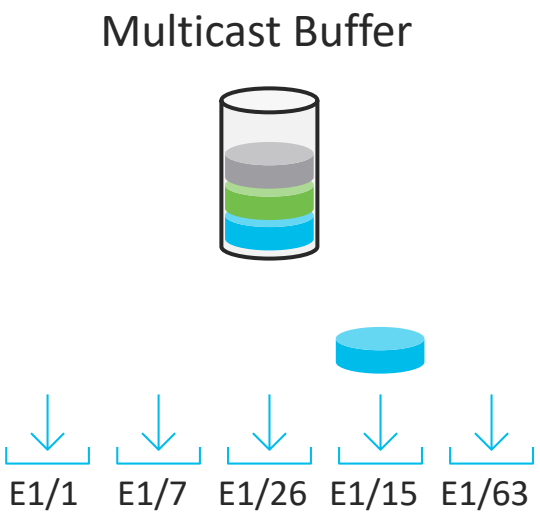
- Packets are stored in the multicast buffer in the ASIC
- If multiple packets are processed, all stored in the same buffer
- Packets are replicated, by reading packets from multicast buffer

Why is this happening in the ASIC?



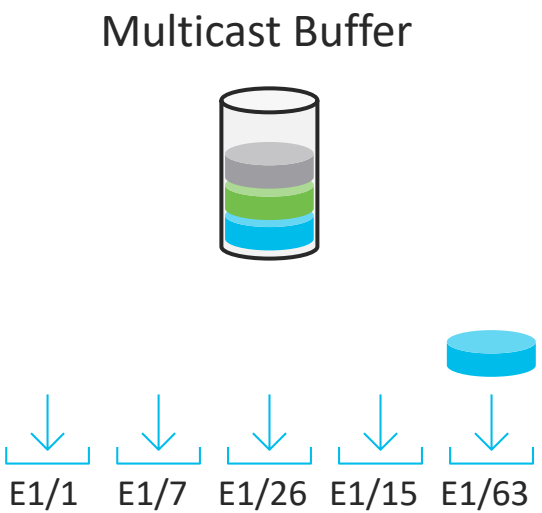
- Packets are stored in the multicast buffer in the ASIC
- If multiple packets are processed, all stored in the same buffer
- Packets are replicated, by reading packets from multicast buffer

Why is this happening in the ASIC?



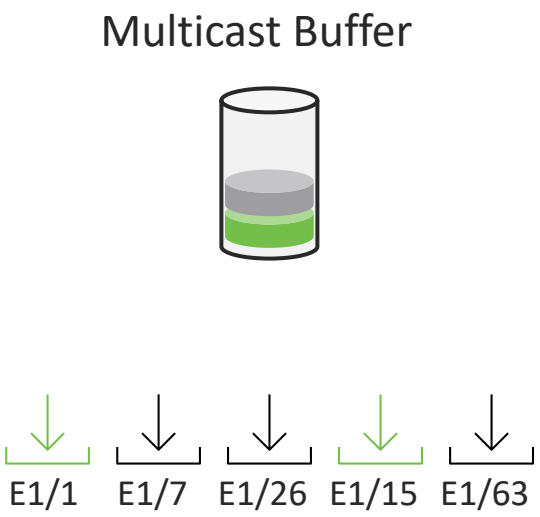
- Packets are stored in the multicast buffer in the ASIC
- If multiple packets are processed, all stored in the same buffer
- Packets are replicated, by reading packets from multicast buffer

Why is this happening in the ASIC?



- Packets are stored in the multicast buffer in the ASIC
- If multiple packets are processed, all stored in the same buffer
- Packets are replicated, by reading packets from multicast buffer

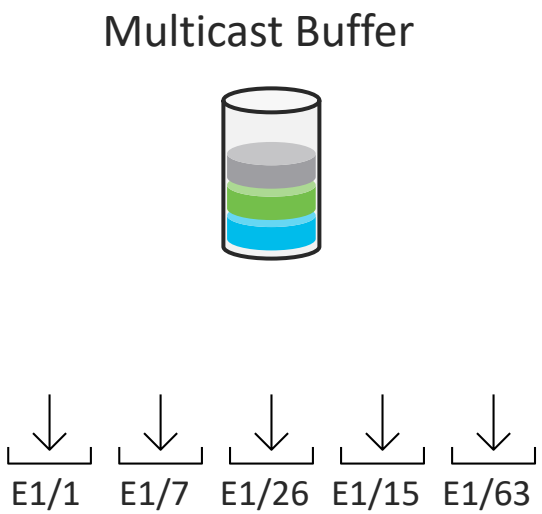
Why is this happening in the ASIC?



- Packets are stored in the multicast buffer in the ASIC
- If multiple packets are processed, all stored in the same buffer
- Packets are replicated, by reading packets from multicast buffer
- After last port sends out packet, it is deleted from buffer

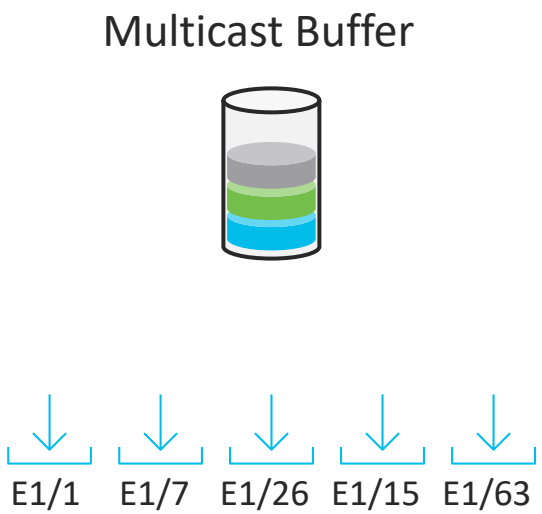
Can FPGA help?

Can FPGA help?



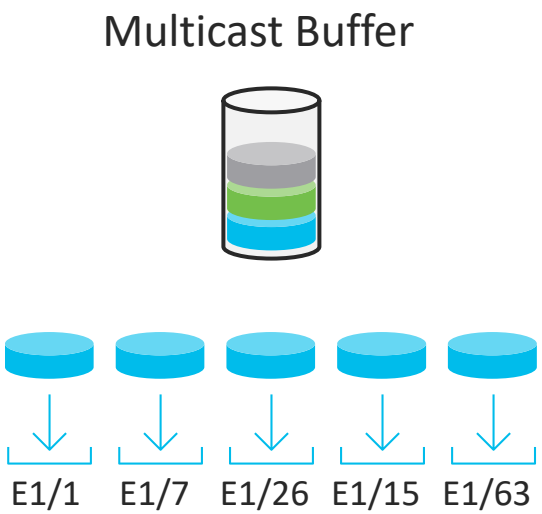
- In FPGA based network switch, multicast replication is parallel

Can FPGA help?



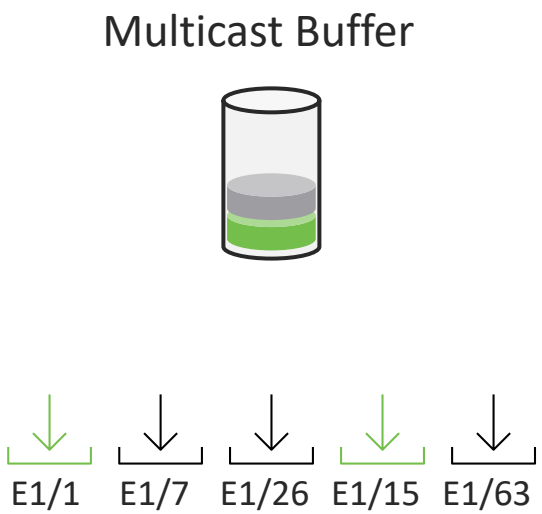
- In FPGA based network switch, multicast replication is parallel
- All ports members of multicast group send packet at same time

Can FPGA help?



- In FPGA based network switch, multicast replication is parallel
- All ports members of multicast group send packet at same time

Can FPGA help?

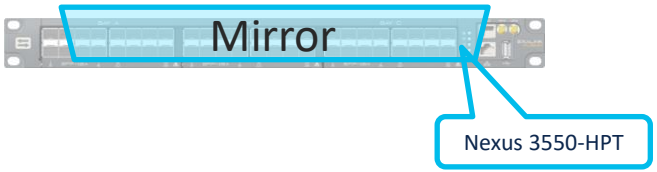


- In FPGA based network switch, multicast replication is parallel
- All ports members of multicast group send packet at same time

How was the delay measured?

How was the delay measured?

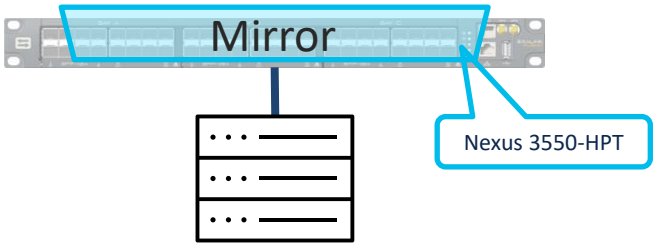
- Higher precision latency measure:
  - Nexus 3550-F HPT – performs ingress time stamping at 70ps* precision, and mirroring



*Not a STAC benchmark

How was the delay measured?

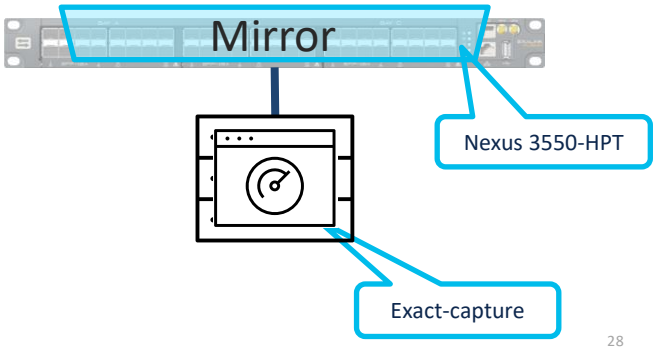
- Higher precision latency measure:
  - Nexus 3550-F HPT – performs ingress time stamping at 70ps* precision, and mirroring



*Not a STAC benchmark

How was the delay measured?

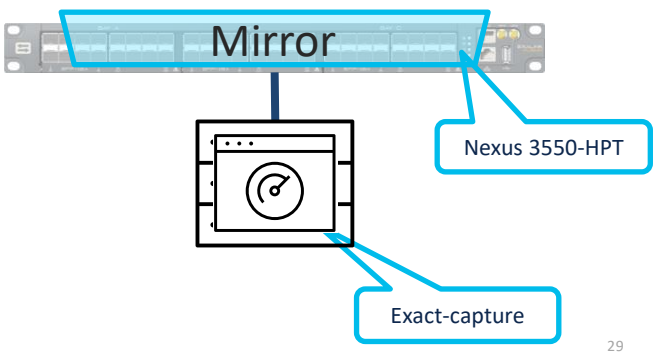
- Higher precision latency measure:
  - Nexus 3550-F HPT – performs ingress time stamping at 70ps* precision, and mirroring
  - Exact-capture tool set - open-source software to analyze time stamps



*Not a STAC benchmark

How was the delay measured?

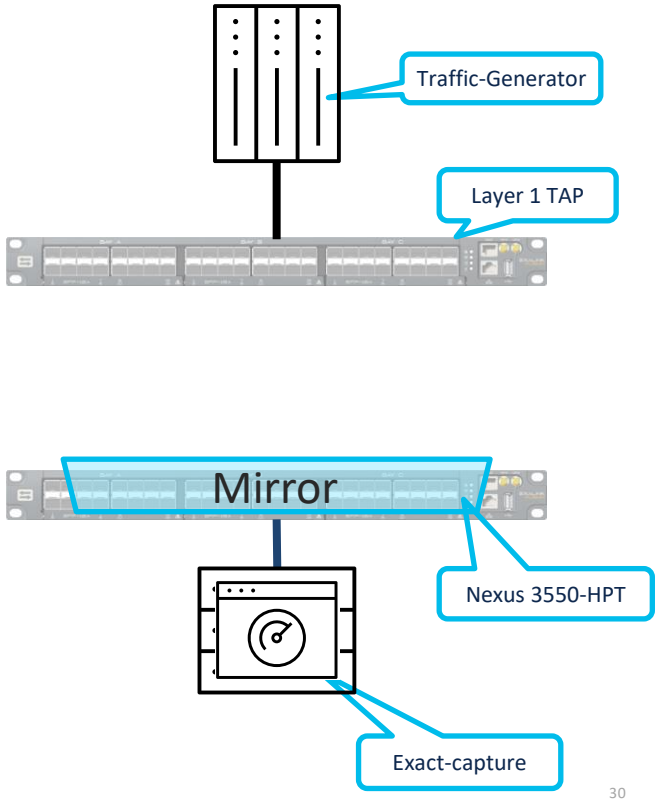
- Higher precision latency measure:
  - Nexus 3550-F HPT – performs ingress time stamping at 70ps* precision, and mirroring
  - Exact-capture tool set - open-source software to analyze time stamps
  - Traffic generator, or another source of multicast traffic



*Not a STAC benchmark

How was the delay measured?

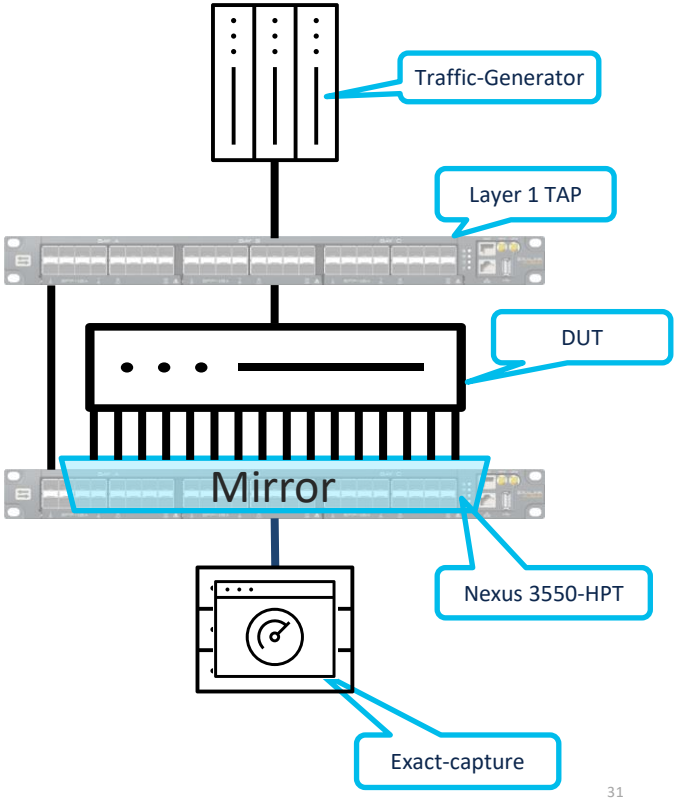
- Higher precision latency measure:
  - Nexus 3550-F HPT – performs ingress time stamping at 70ps* precision, and mirroring
  - Exact-capture tool set - open-source software to analyze time stamps
  - Traffic generator, or another source of multicast traffic
  - Layer 1 TAP to distribute source of traffic to two different ports



*Not a STAC benchmark

How was the delay measured?

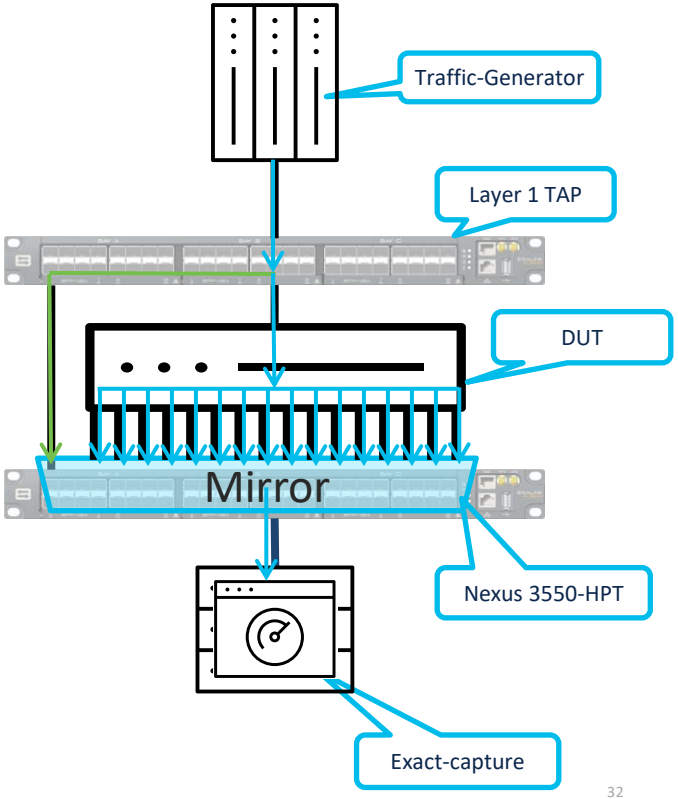
- Higher precision latency measure:
  - Nexus 3550-F HPT – performs ingress time stamping at 70ps* precision, and mirroring
  - Exact-capture tool set - open-source software to analyze time stamps
  - Traffic generator, or another source of multicast traffic
  - Layer 1 TAP to distribute source of traffic to two different ports
  - DUT on what latency and fairness is performed



*Not a STAC benchmark

How was the delay measured?

- Higher precision latency measure:
  - Nexus 3550-F HPT – performs ingress time stamping at 70ps* precision, and mirroring
  - Exact-capture tool set - open-source software to analyze time stamps
  - Traffic generator, or another source of multicast traffic
  - Layer 1 TAP to distribute source of traffic to two different ports
  - DUT on what latency and fairness is performed
  - Traffic is sent to DUT, so latency of distribute, traffic latency is measured.



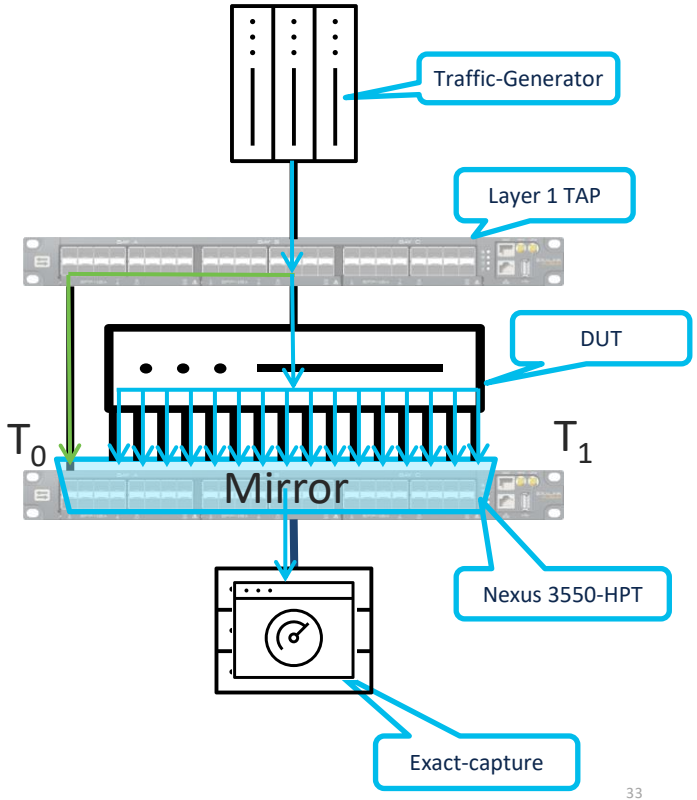
*Not a STAC benchmark

How was the delay calculated?

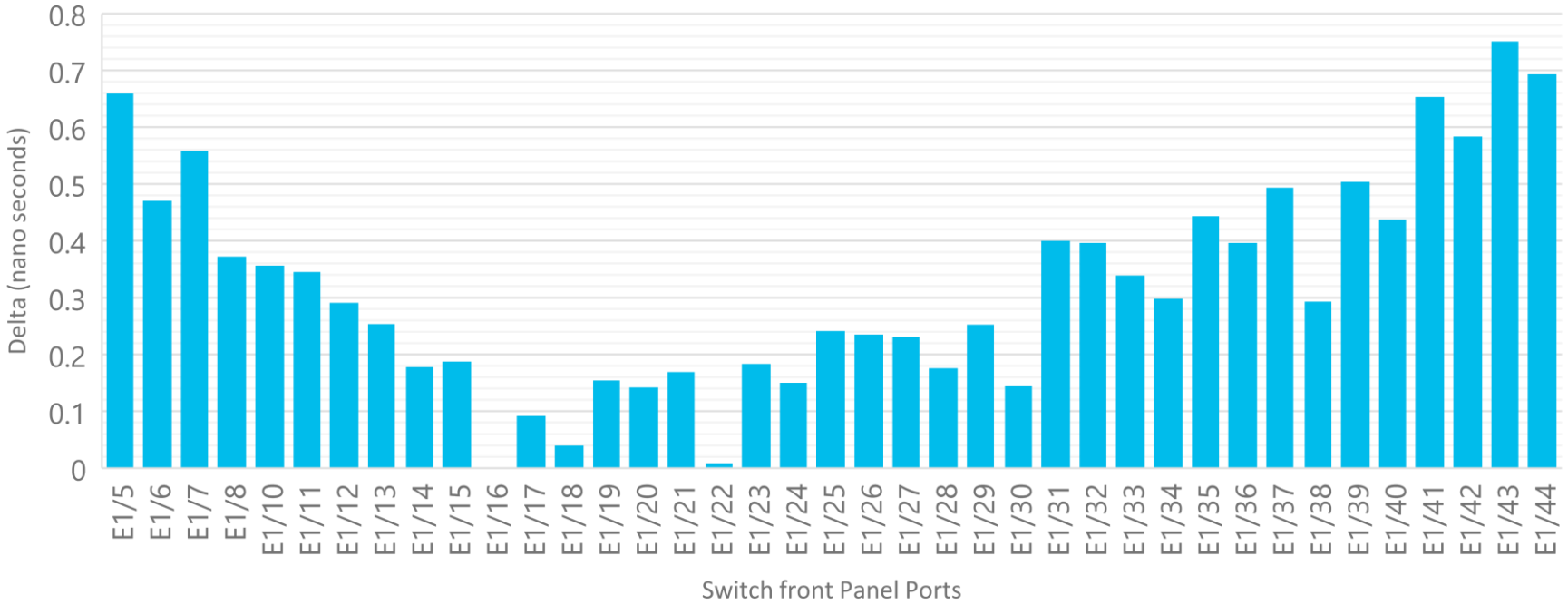
- Nexus 3550-F HPT time stamps packet at ingress port:
  - Time T_0 is reference time, where T_1 is time with addition of DUT latency
  - T_1 is produced per port, T_{1P1}, T_{1P2}…
  - Traffic is mirrored toward Exact-capture
  - Exact-capture, processes time stamps and provides per port latency
  - By processing per port latency further, delay can be calculated as latency delta between ports

$Latency_{P1} = T_{1P1} - T_0$

$Delta\ between\ ports = Latency_{P1} - Latency_{P2}$



Nexus 3550-T Market Data Fairness*

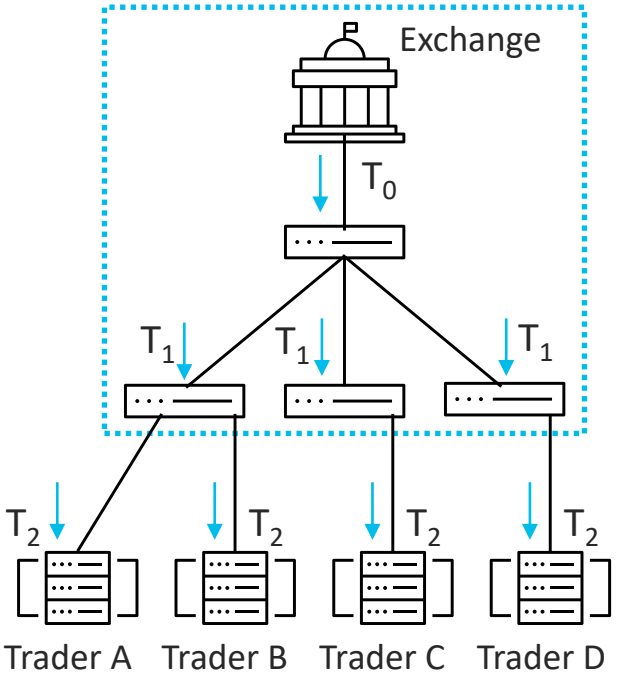


Per port delay from fastest port in this sample – all ports are inside of 1ns

*Part of future NXOS software release

Not STAC Benchmark

Solution – Market Data Distribution with FPGA



- With Cisco FPGA based network switches, distribution is happening with minimal delay
- Each network node, treat ports fairly, so each port will get packet at the same time
- Even with multiple hops in the network each trader will receive market data at the same time as others

Nexus 3550-T – Runs NX-OS

Cisco NX-OS support

Same NXOS CLI
Same APIs
Support in NDFC

Nexus 3550-T – Runs NX-OS

Cisco NX-OS support

Same NXOS CLI
Same APIs
Support in NDFC

Low Latency Layer 2 and 3

Port to port latency 95-160 nano seconds*

***Not STAC Benchmark**

Nexus 3550-T – Runs NX-OS

Cisco NX-OS support	Same NXOS CLI Same APIs Support in NDFC
Low Latency Layer 2 and 3	Port to port latency 95-160 nano seconds*
FPGA	Xilinx Ultrascale+ VU35P-3 FPGA with 8GB HBM2

***Not STAC Benchmark**

Nexus 3550-T – Runs NX-OS

Cisco NX-OS support	Same NXOS CLI Same APIs Support in NDFC
Low Latency Layer 2 and 3	Port to port latency 95-160 nano seconds*, 25G capable
FPGA	Xilinx Ultrascale+ VU35P-3 FPGA with 8GB HBM2
Custom FDK	Design FPGA application on the switch

***Not STAC Benchmark**

The bridge to possible