Switch Selection (and buffer sizing)

Joel Jaeggli | 03-05-2019

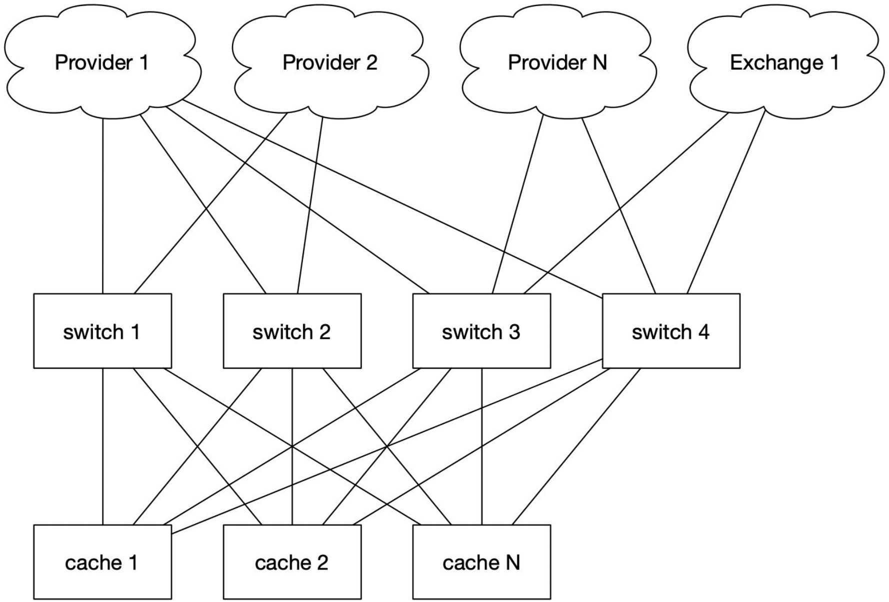
Fastly Backstory

- Founded 2011
- Original topology is single cache directly attached to transit and exchange providers.
- Fastly Network Architecture is very cache-centric
  - Caches carry full routing tables
  - Caches make exit selection decisions.
  - Switches serve as mediation layer / multiplexor between carriers /exchanges and switches.

fastly

Topology

- Simplified Fastly topology
- For pops sized from 4-32 caches these were 48 port 1ru 10Gb/s switches (Trident+, Trident2, FM6000)



fastly

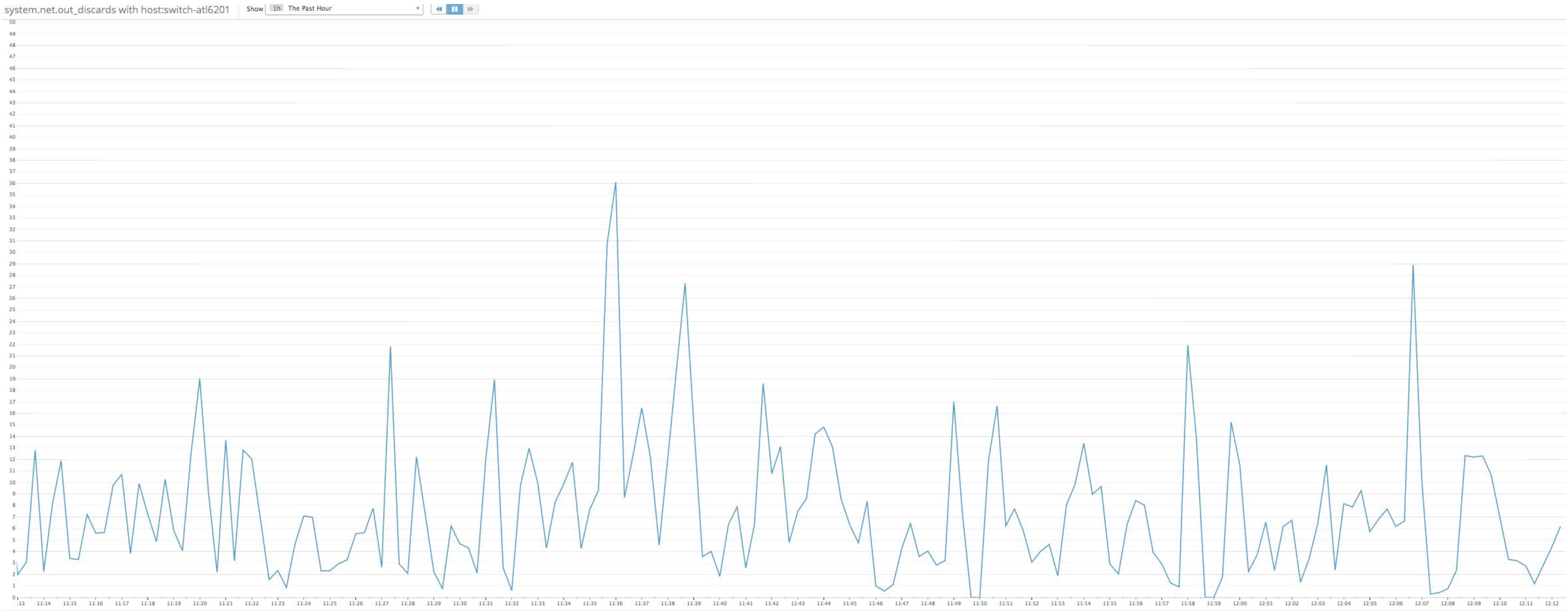
Historical switches

- Single ASIC per device
- Cut-through forwarding
- Very low latency (350ns for some FM6000 variants)
- All ports run at 10Gb/s
  - even 40 Gb/s ports are configured as 4 x 10 Gb/s
- Small shared memory buffer
  - 8MB in T+, 12MB in T2, 7.5MB in FM6000

fastly

Observing buffering behavior indirectly

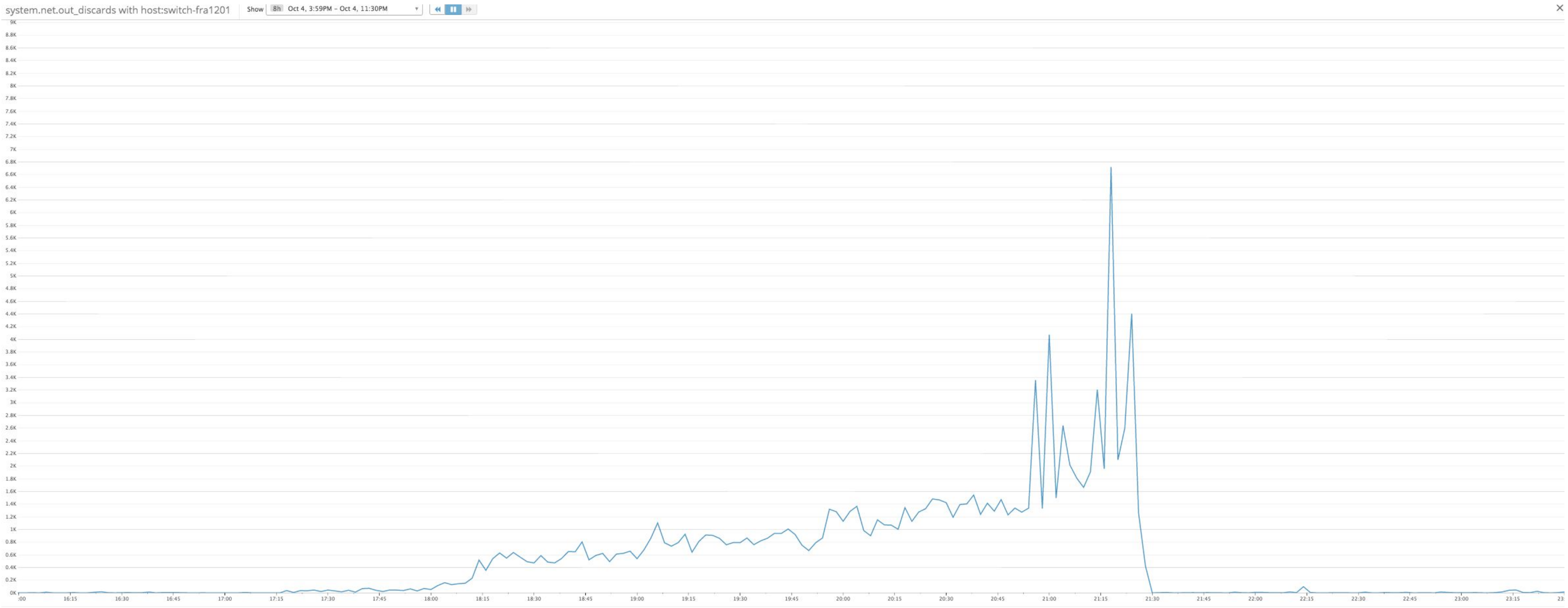
- Drops on T2, no sustained congested ports. (15s sample interval)



fastly

Observing buffering behavior indirectly

- Drops under duress (FM6000) (across all output ports)



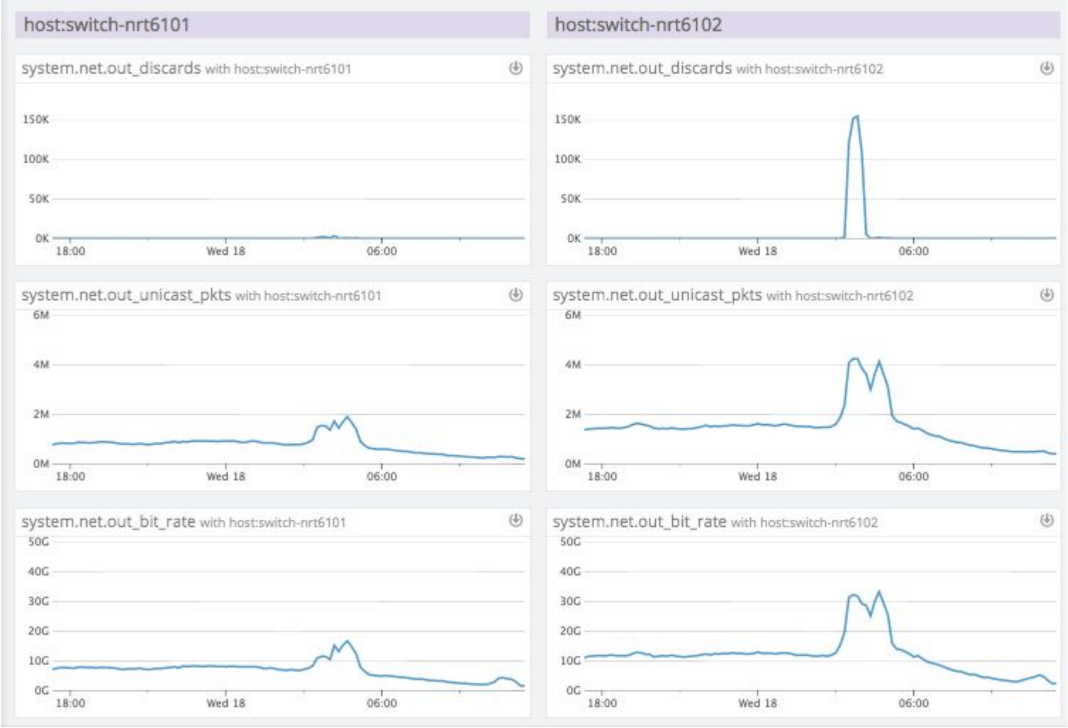
What happened there?

- Traffic ramped up to in total around 50Gb/s.
- However only a single port was congested.
- Because of the small shared memory pool all enqueued packets on the switch are subject to discard due to one congested 10Gb/s port.
- Classic TCP incast problem.
- QOS policy subsequently implemented favors discarding bulk precedence traffic (in this case HLS streaming and large objects) rather than cache-cache traffic

fastly

Another example, two switch exposed to a single event, one with a congested port.

Where we have two switches in the same pop exposed to high demand only one is discarding anything of substance.



fastly

In general fairly happy with these platforms, however:

- Would like to be less exposed to congestion events that impact only 1 port.
- 100Gb/s is coming along.
- Many more mixed rate interfaces present
  - 100Gb/s provider circuits
  - 25Gb/s host interfaces
  - 10Gb/s peering circuits
  - Cut-through forwarding no-longer possible

100Gb/s ASICs

- Broadcom Tomahawk, feature reduced 100Gb/s ASIC with 16MB of buffer split between 4 forwarding cores.
  - 32 x 100Gb/s ports per ASIC
  - seems to be heading in the wrong direction

- Dune Arad / Jericho on the other hand
  - cell forwarder rather than a cut-through ethernet switch
  - 8 - 10 ports exposed per ASIC
  - 4GB of external port buffer per ASIC
  - much slower / higher latency (3.5usec minimum) but better scale properties

Jericho VOQ buffers

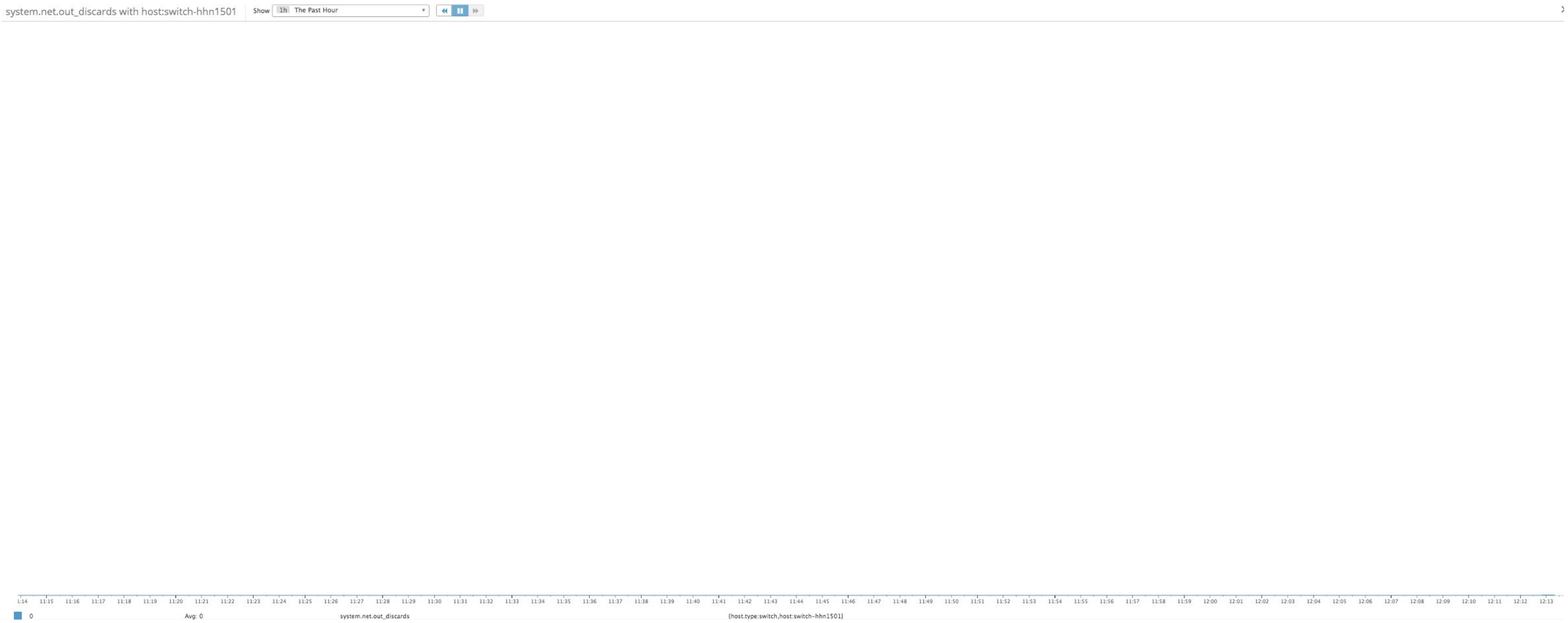
- Can be outlandish
  - 500MB per port
  - … or 40ms per port
- Requires policing if you have clear ideas about queue depth
- No single port is ever going to soak up the whole buffer

Tail-Drop thresholds configuration:

Speed	DropPrec	traffic-class	MaxQueSize (bytes)	MaxQueBuffSize (buffers)
100Mbps	Normal	-	1310720 (1.25 MB)	5000
100Mbps	Cpu	-	1415577 (1.35 MB)	6000
1Gbps	Normal	-	13107200 (12.50 MB)	12500
1Gbps	Cpu	-	14155776 (13.50 MB)	13500
10Gbps	Normal	-	52428800 (50.00 MB)	50000
10Gbps	Cpu	-	53477376 (51.00 MB)	51000
25Gbps	Normal	-	131072000 (125.00 MB)	125000
25Gbps	Cpu	-	133693440 (127.50 MB)	127500
40Gbps	Normal	-	209715200 (200.00 MB)	200000
40Gbps	Cpu	-	213909504 (204.00 MB)	204000
50Gbps	Normal	-	262144000 (250.00 MB)	250000
50Gbps	Cpu	-	267386880 (255.00 MB)	255000
100Gbps	Normal	-	524288000 (500.00 MB)	500000
100Gbps	Cpu	-	534773760 (510.00 MB)	510000

Large VOQ, queue drops

Pretty much none.



What to make of this?

- Impact on traffic of microbursts on very small buffer devices is hard to quantify in the field.
- RTT derived buffering assumptions produce queue depths incompatible with low latency data delivery.
- Switch architectures vary greatly and buffer sizes along with them.
- Appearance of mutually incompatible approaches exist in the same marketplace and from the same vendors!
- Better methodology required.

Thank You

sales@fastly.com