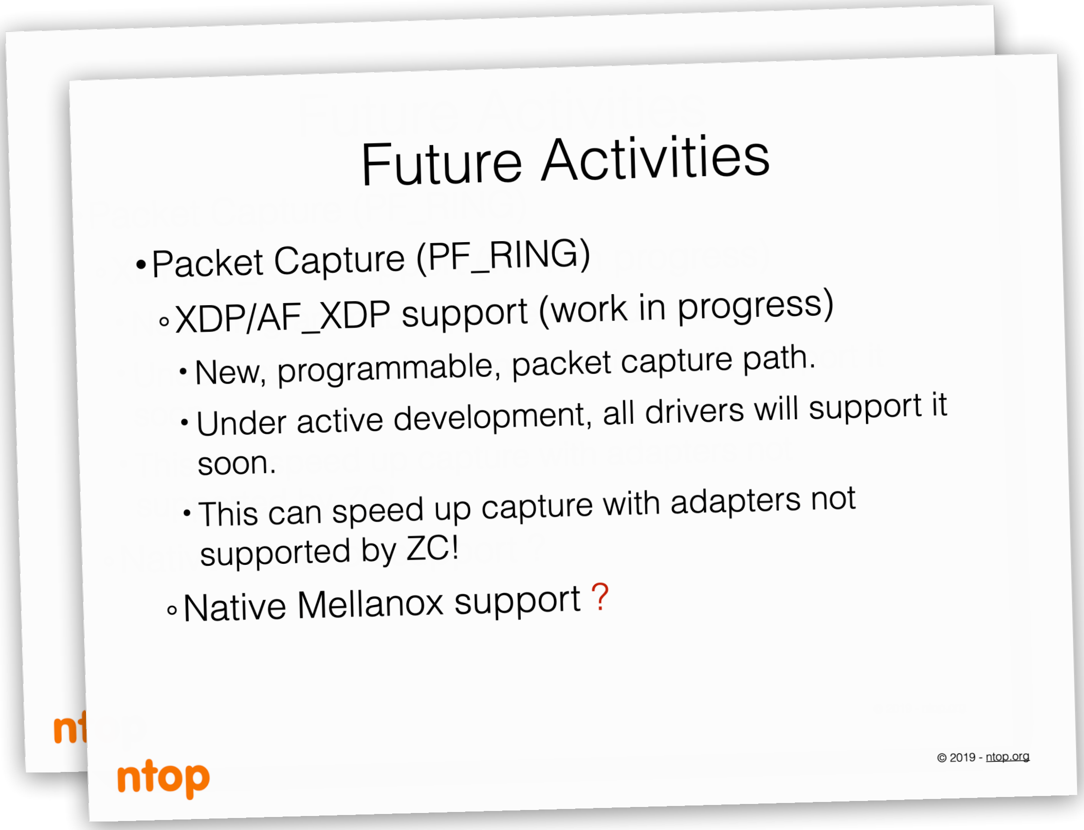
PF_RING & n2disk Since Last ntopConf

Alfredo Cardigliano
cardigliano@ntop.org

Last time we met...

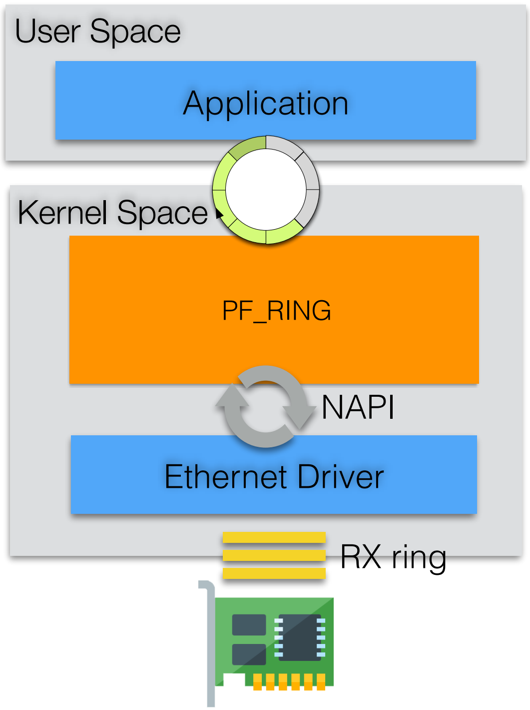


What's New In PF_RING

PF_RING

- Introduced in 2004 for improving the performance of network monitoring applications, by providing packet capture acceleration

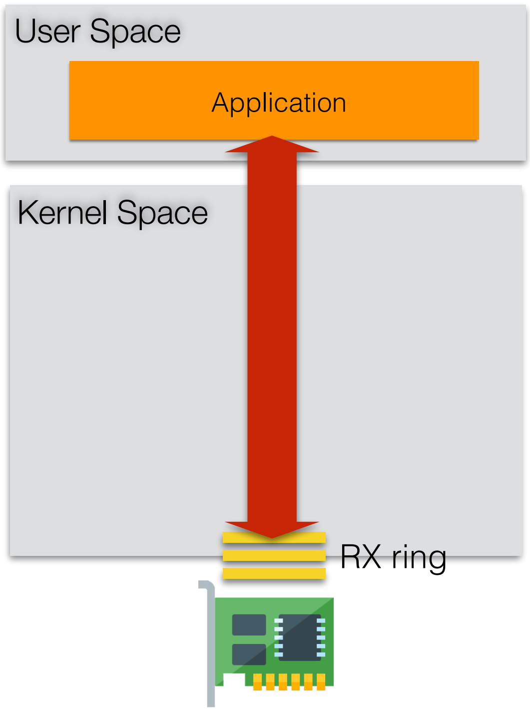
- PF_RING offers on commodity hardware (a standard PC with commodity Network adapters) the ability to receive and transmit at high speed



PF_RING ZC

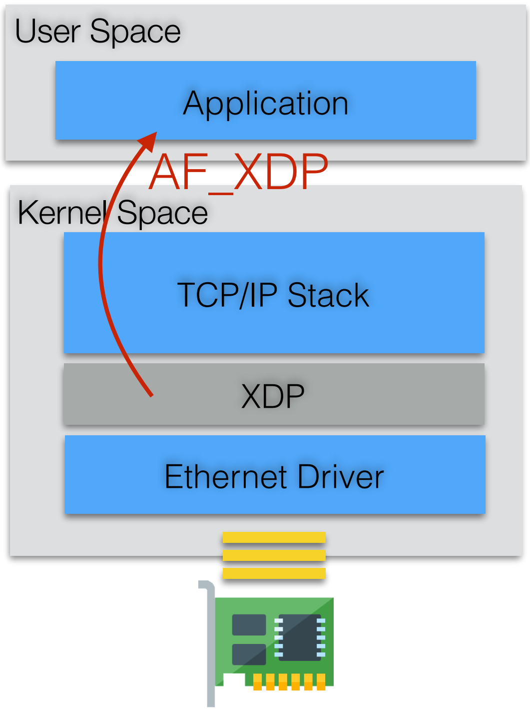
- Wire-rate packet capture up to 100 Gbit using kernel-bypass zero-copy drivers with commodity adapters (e.g. Intel)

- Support for many (almost all) specialized FPGA adapters on the market (Napatech, Silicom Fiberblaze, Accolade, etc.)



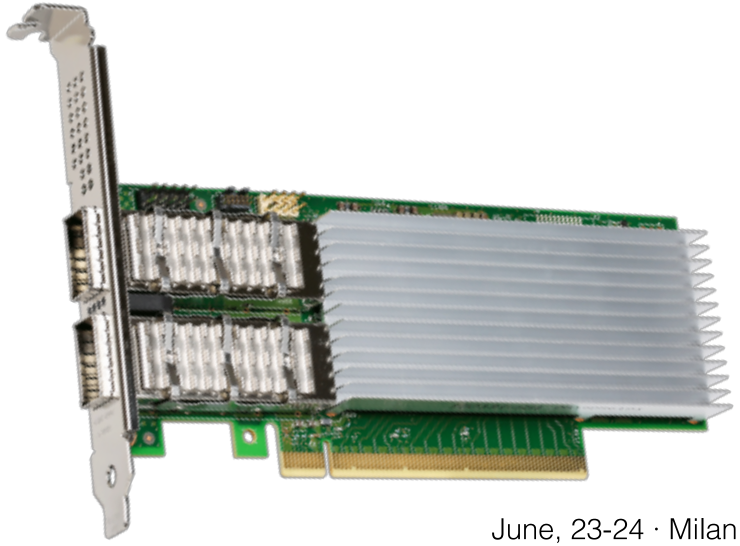
PF_RING Over XDP

- XDP (eXpress Data Path) is a new layer in the Linux kernel before the network stack
- Not kernel bypass: data-plane inside the kernel, programmable using eBPF programs
- AF_XDP is the socket used to deliver packets to userspace
- PF_RING 8 introduces an optimized support for zero-copy/batch capture using AF_XDP



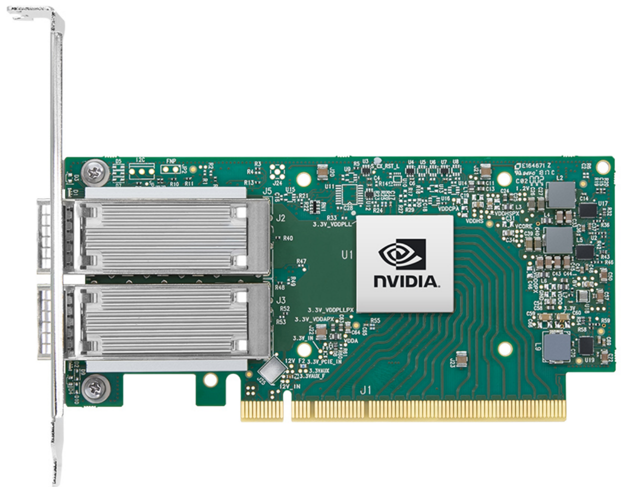
Intel E810 "Columbiaville"

- New PF_RING ZC "ice" driver for the new family of Intel 100 Gbit Ethernet adapters (ice)
  - This replaces "fm10k" Intel 100 Gbit adapters
- Supported link speed: 10/25/50/100 Gbit
- Capture performance: 25 Mpps per queue/core



Mellanox Connect-X

- New PF_RING ZC driver for Mellanox (NVIDIA) Ethernet adapters (Connect-X 4/5/6)
- Supported link speed: 10/25/40/50/100/200 Gbit
- Support for many RSS queues (multithread applications)
- Flexible hardware filtering
- Hardware timestamping



Mellanox Performance

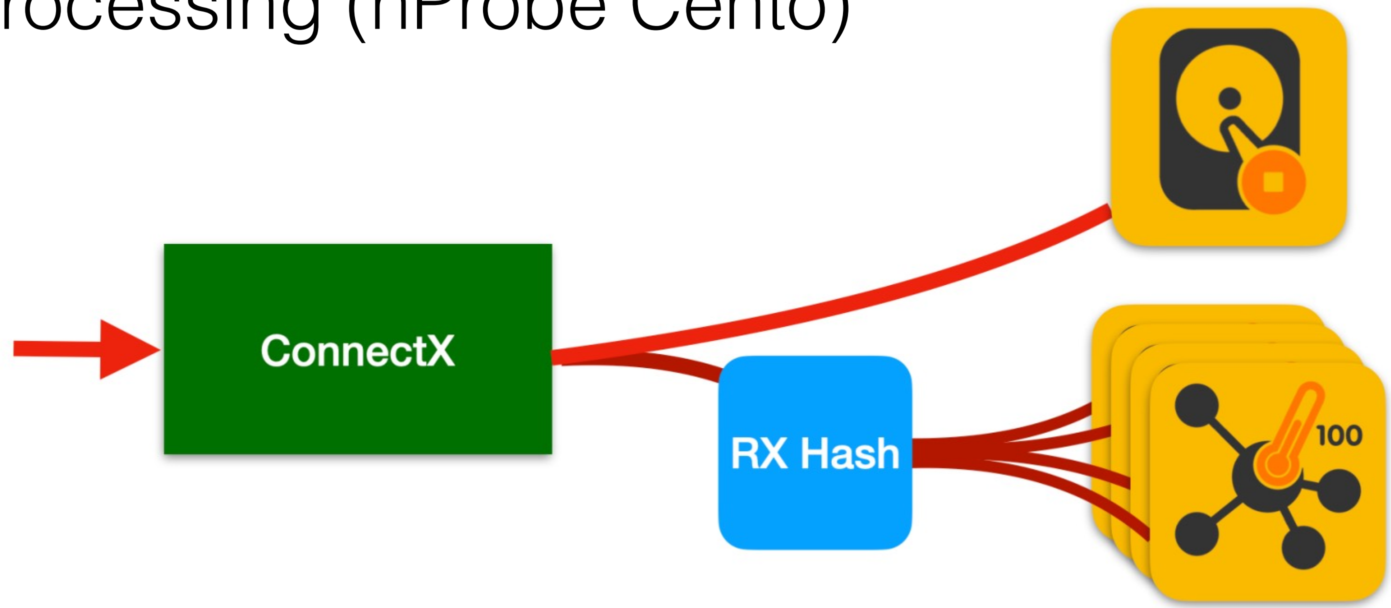
- Capture performance:
  - 32 Mpps on a single core (20 Gbps with worst-case 60-byte packets, 40 Gbps with an avg packet size of 128 bytes)
  - Full 100 Gbps using RSS
- Application performance (nProbe Cento):
  - 100 Gbps worst-case traffic (small packet size) using 16 cores
- Tested with Mellanox ConnectX-5 on Intel Xeon Gold 16-cores @ 2.2/3.5 GHz

Load-Balancing / Duplication

- As opposite to ZC drivers for Intel, access to the device is non exclusive on Mellanox, even in zero-copy kernel-bypass mode
- It is possible to capture traffic from multiple applications (traffic <u>duplication</u>)
- Different load-balancing (RSS) configuration for each application

Load-Balancing / Duplication

- Example
  - Full traffic to a single data stream for traffic recording (n2disk)
  - Load-balancing to N streams/cores for processing (nProbe Cento)



Hardware Filtering

- High number of hardware rules (64K on Mellanox ConnectX-5)
- Flexible rules: compose rules by specifying which packet headers (protocol, src/dst IP, src/dst port, etc) and masks, should be used to match the rule
- Drop or pass actions (with default accept or deny)
- Rules priority support, also across applications

n2disk: How To Build a 100 Gbit Network Recorder

Continuous Recording

- Going back in time and drilling down to the packet level could be crucial to find the exact network activity that caused an issue.

- In most cases it's not possible to predict when a network event occurs, we need to record traffic until the problem occurs.

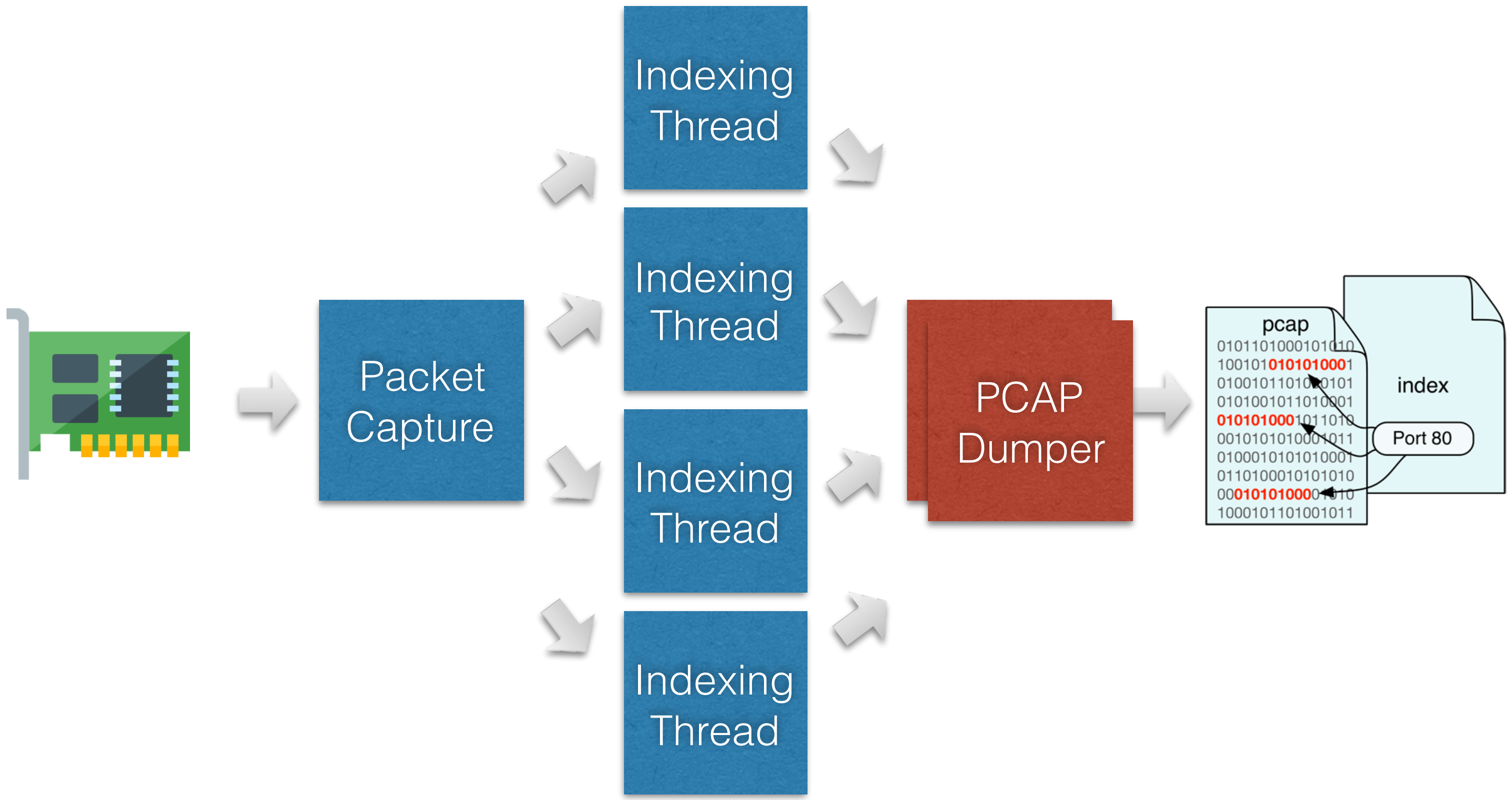
- Large companies are often protected by firewalls and IDSs (Intrusion Detection Systems). Those security tools do not keep traffic history but just log security events.

n2disk

- 1/10/40/100 Gbit traffic recorder
- It relies on PF_RING for capturing and processing traffic with no packet loss up to 100 Gbps sustained
- It uses the industry standard PCAP file format to dump packets into files
- Hardware timestamps with nanosecond accuracy (with supported adapters)
- Full packets are stored and indexed to enable on-demand retrieval (BPF)

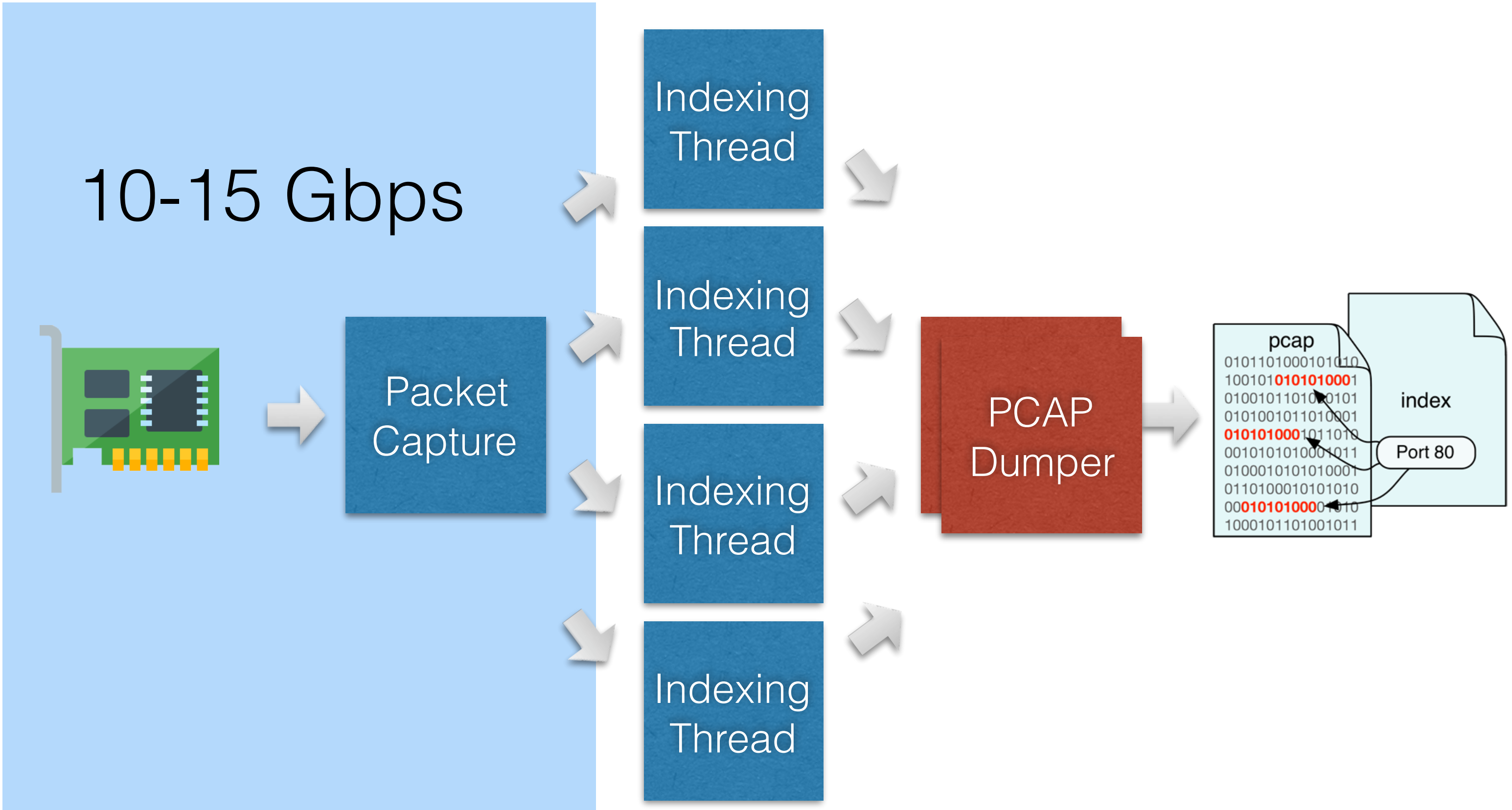
Technology

- Multithreaded packet processing architecture



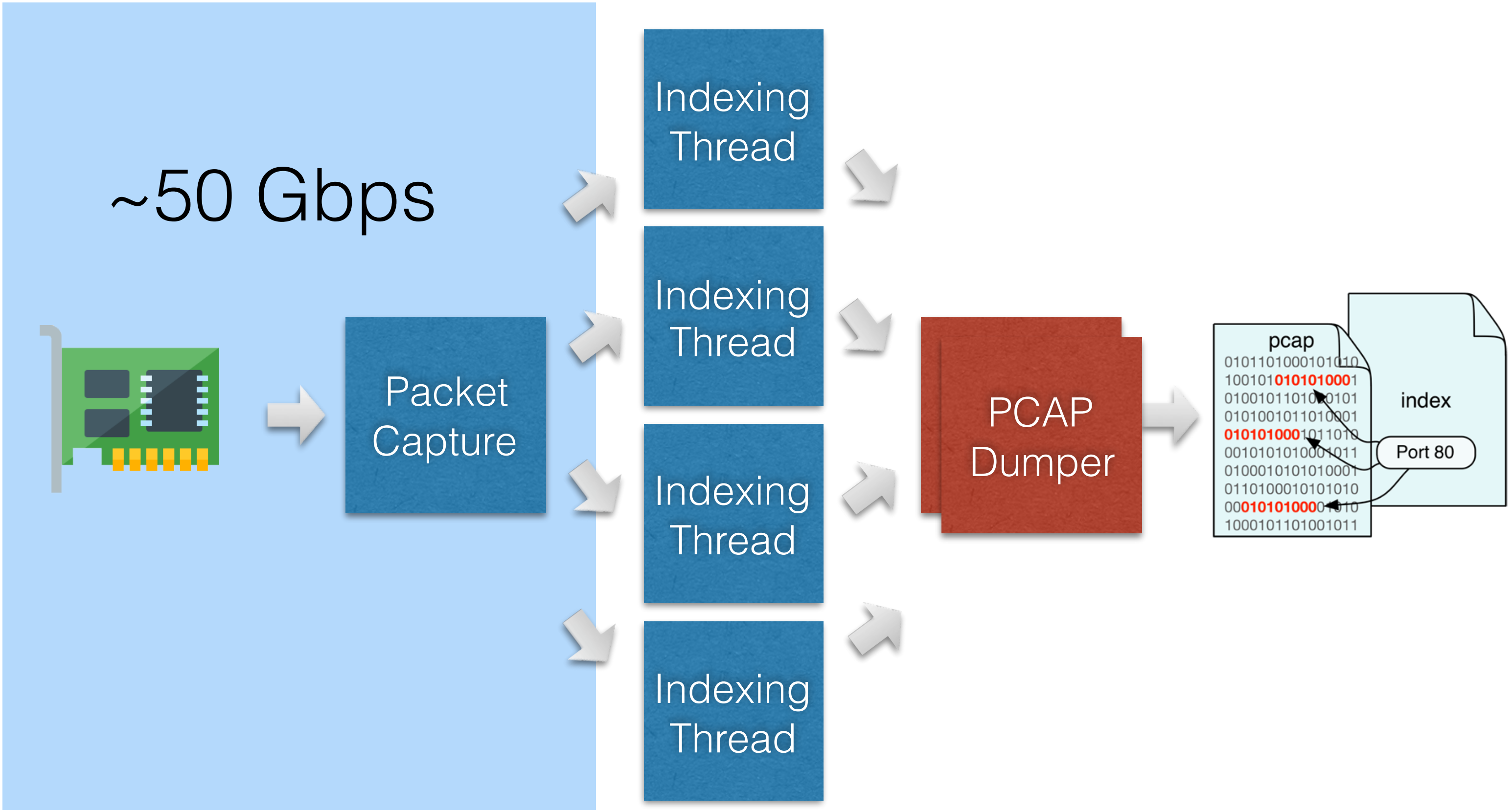
Capture Performance - ASIC

- Commodity ASIC NICs (e.g. Intel) work per-packet (many transactions on the PCIe bus, single packets are moved off the adapter)



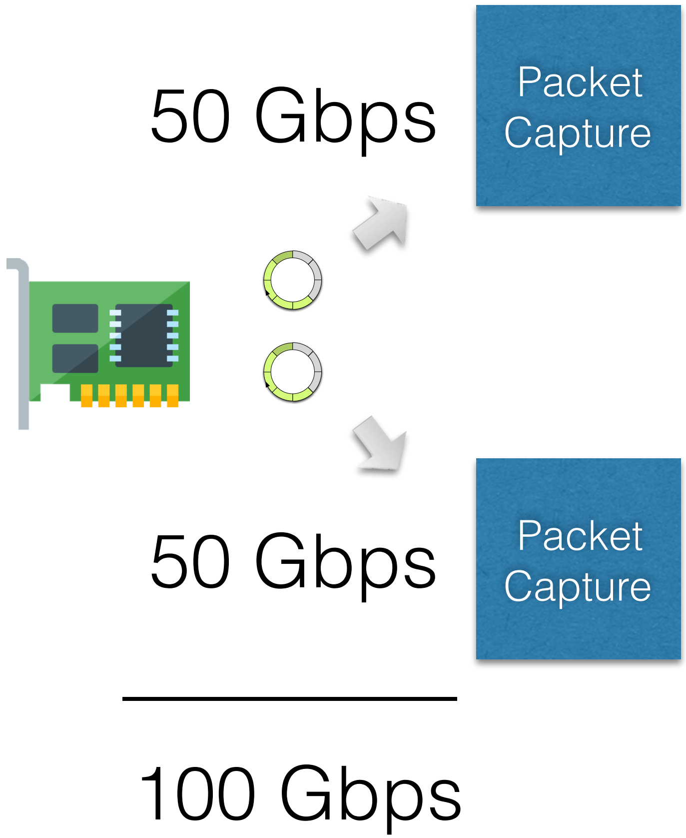
Capture Performance - FPGA

- FPGA NICs support block mode (less pressure on the PCIe bus, data blocks are moved off the adapter)



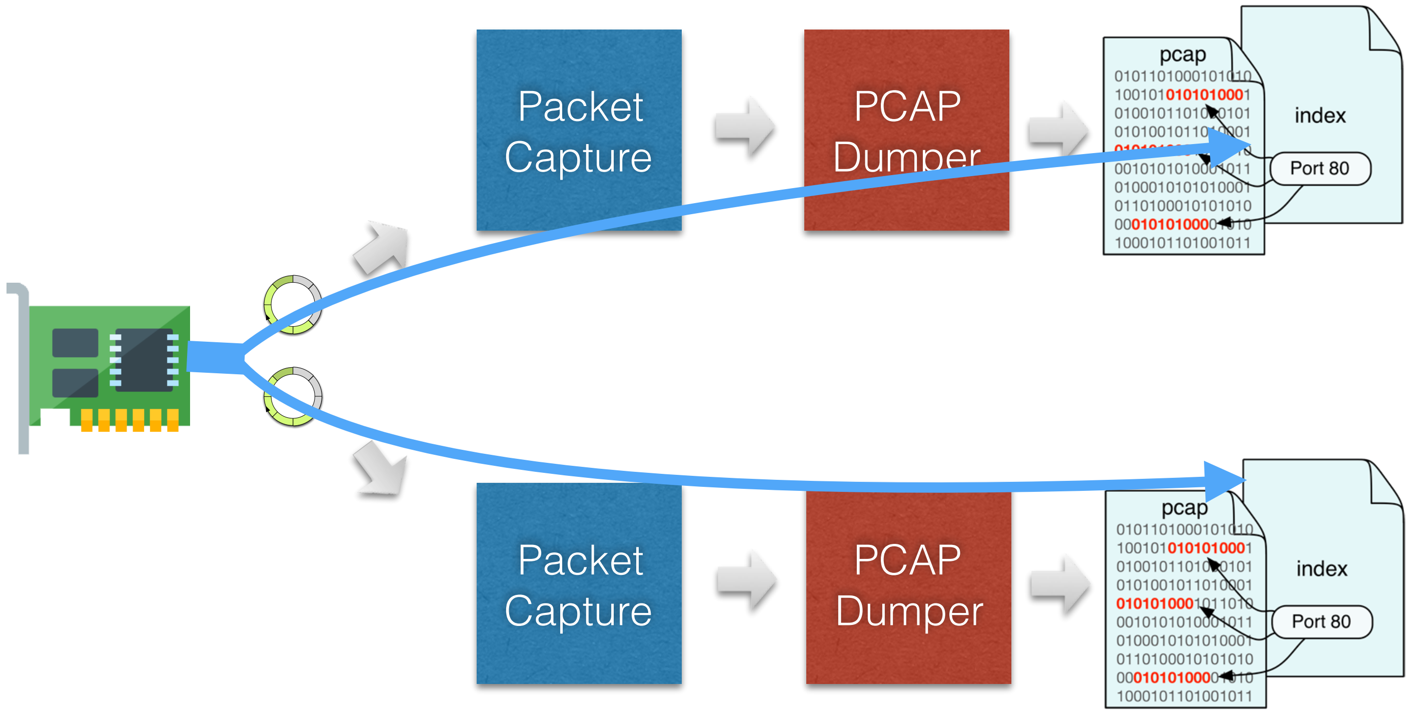
Scale Capture Performance

- RSS is usually used to load balance incoming traffic and spread it across multiple queues where cores operate in parallel



RSS Drawbacks

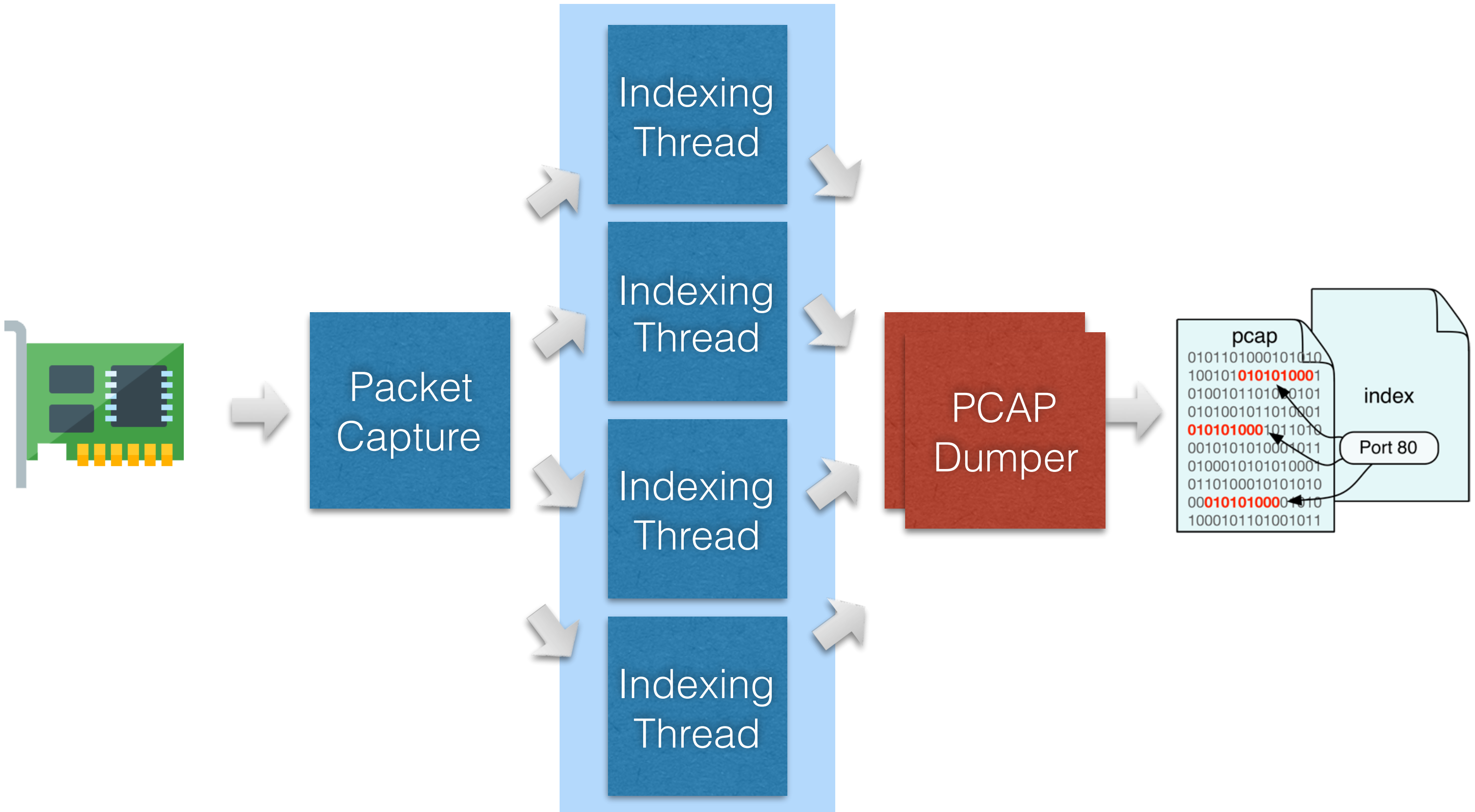
- RSS shuffles ingress traffic, loosing the order of network packets on the wire, required to provide evidence of a Network issue



- However, hardware timestamps (when available) can be used to sort packets at extraction time

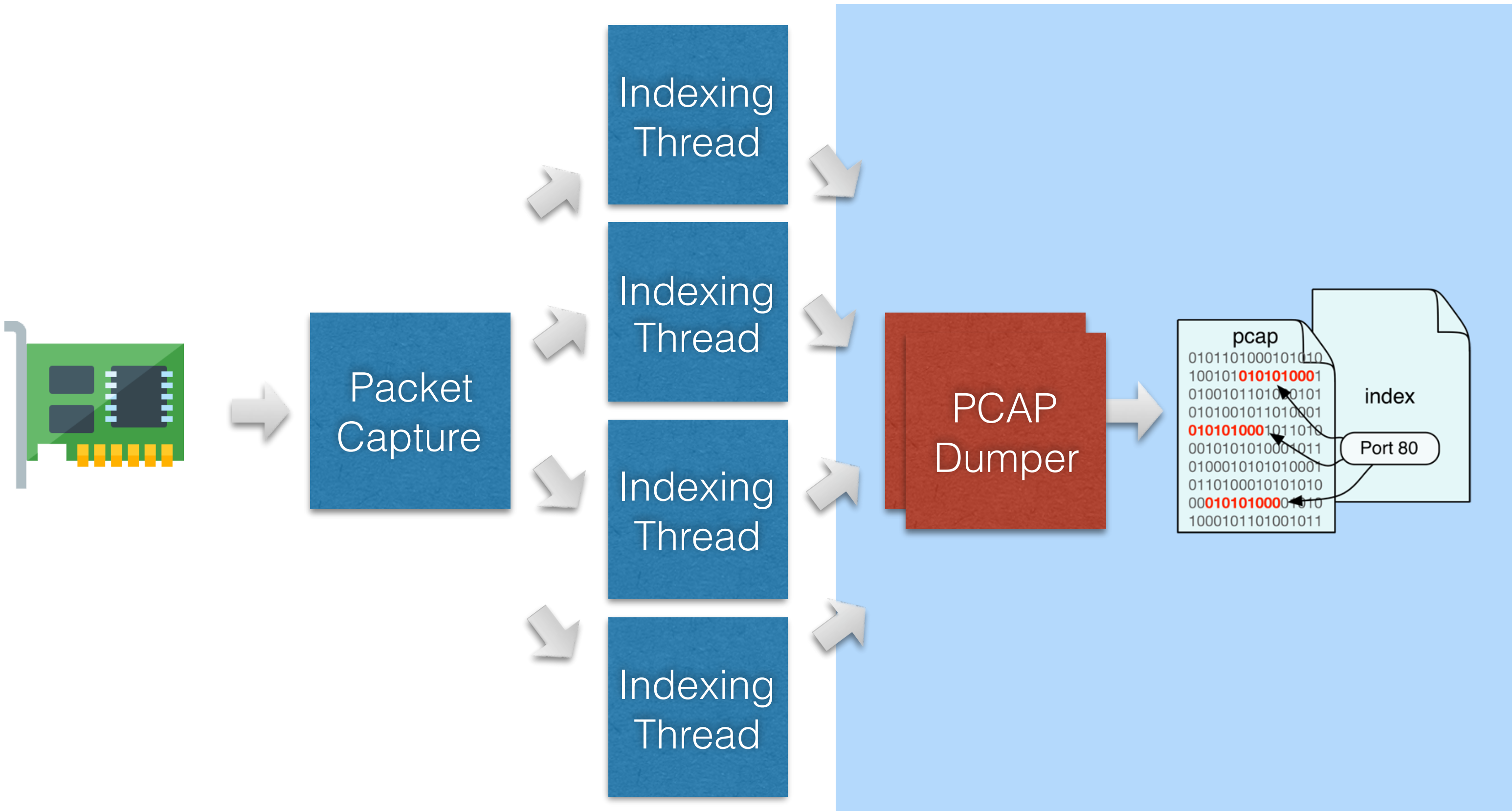
Index Performance

- A single core can process/index 10-15 Gbps (4 cores can handle 50 Gbps)



Dump Performance

- What about the storage?



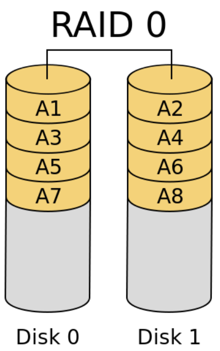
Drives Performance

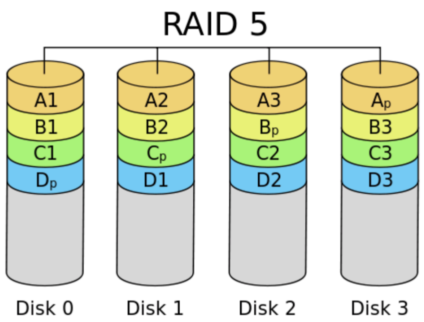
Drive Type	Random IOPS	Sustained Sequential I/O	
SAS/SATA 7,200RPM	70 – 175	100 – 230 MB/s	
SAS 10,000RPM	275 – 300	125 – 200 MB/s	
SAS 15,000RPM	350 – 450	125 – 200 MB/s	1-2 Gbps
2.5" Solid State (SSD)	15,000 – 100,000	110 – 500 MB/s	1-4 Gbps
NVMe PCI-E Solid State (SSD)	70,000 – 625,000	1,100 – 3,200 MB/s	10-20 Gbps

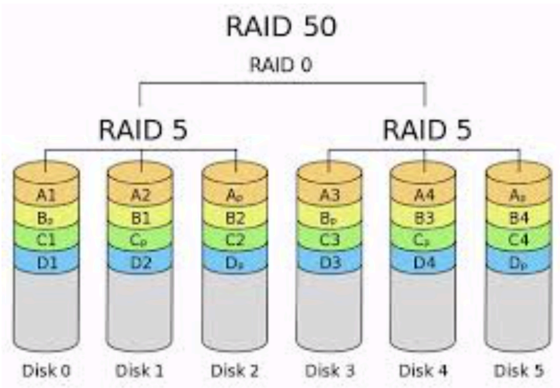
RAID

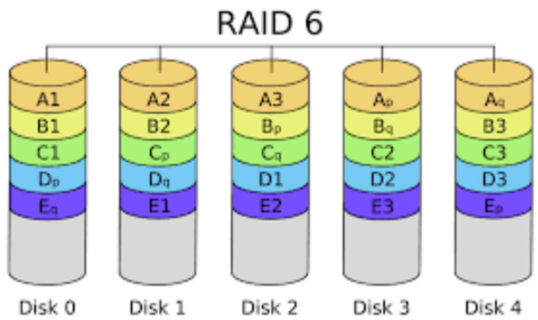
- RAID is a good option for increasing disk bandwidth

- At least 8-10 HDD drives for 10 Gbit when using RAID 0, more drives are required with parity (e.g. RAID 5/50/6)









RAID Performance

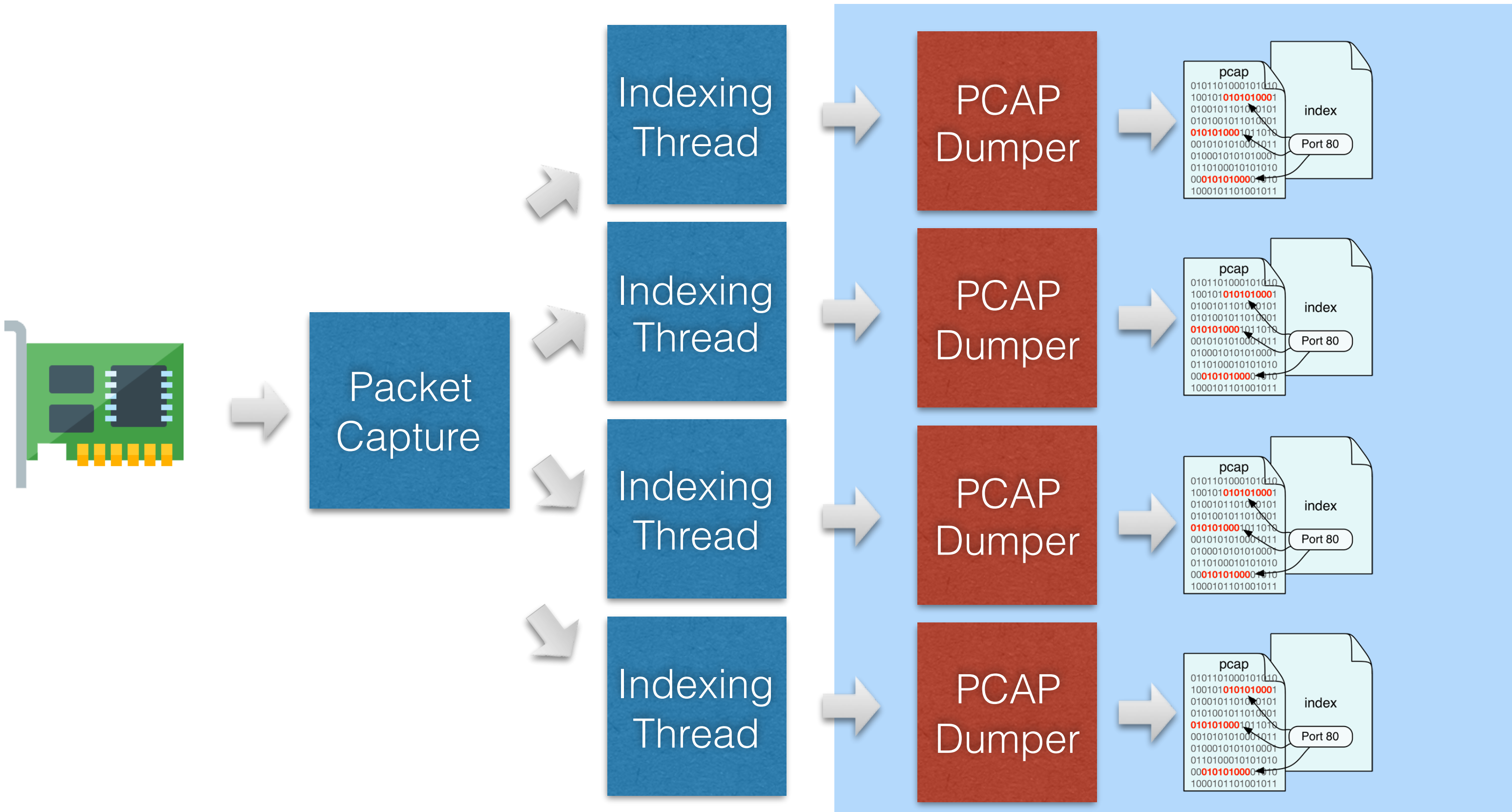
- SATA/SAS 10k/15k RPM HDD drives are a good compromise in terms of price/number
- SSDs should be used when we need to read and write simultaneously to avoid seeking issues
- A RAID controller is usually able to handle ~40 Gbps of write throughput
- Scaling above 40 Gbps requires using multiple RAID controllers :-/

NVMe Disks

- NVMe drives are SSDs directly connected to the PCIe bus
- Pros
  - NVMe are lightfast (~20 Gbps per disk)
  - No need of a RAID controller (they are on the PCIe, a standard SATA/SAS controller cannot drive them)
- Cons
  - A bit expensive, especially those write-intensive
  - Limited number of slots available (usually 10)
- Mandatory at 100 Gbps (~8 drives are enough)

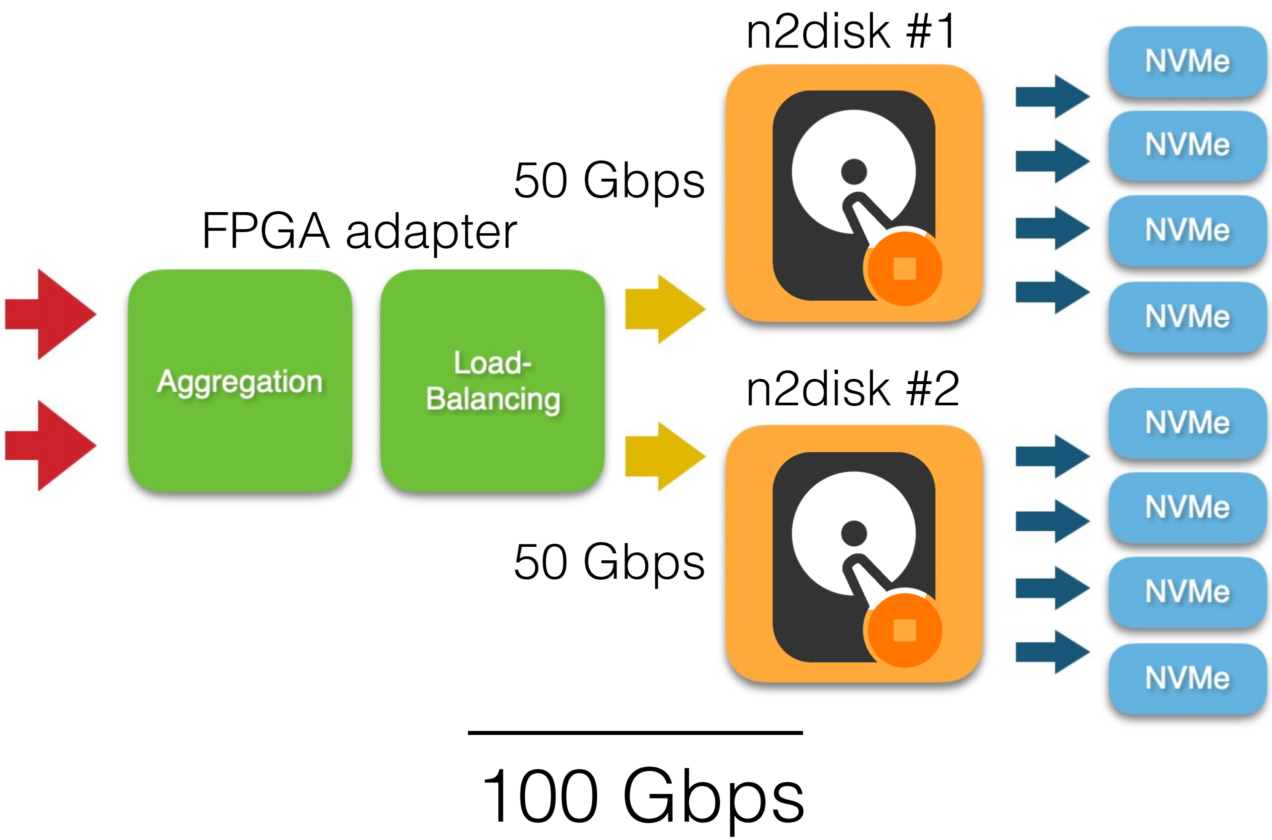
NVMe RAID Emulation

- Multithreaded parallel dump support in n2disk can write in parallel to multiple NVMe disks, emulating a RAID 0



100 Gbps Recording

- Load-balancing to 2 streams
- 2x n2disk instances, able to handle 50 Gbps each
- 8x total NVMe disks

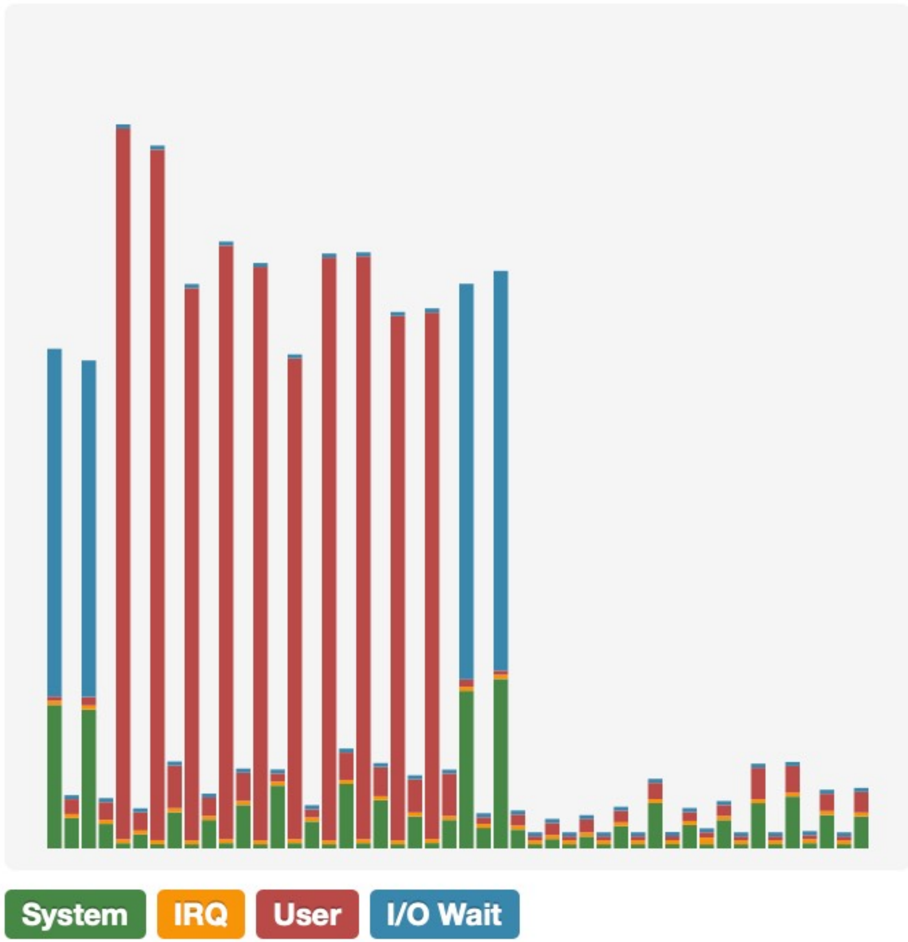


The Recipe for 100 Gbps

- CPU: 16+ Cores 3+ Ghz
- RAM: 64+ GB (or enough to fill all memory channels supported by the CPU)
- Adapter: FPGA with support for segment mode and hardware timestamps (Napatech, Fiberblaze, ..)
- Storage: 8+ NVMe disks (storage size limited by the number of disks available on the box)

CPU Load at 100 Gbps

- CPU cores utilization capturing, indexing and dumping 100 Gbps worst-case traffic (64-byte packets) on a 24-cores system



Thank you