DPDK SUMMIT
Powering the Future of Networking Software

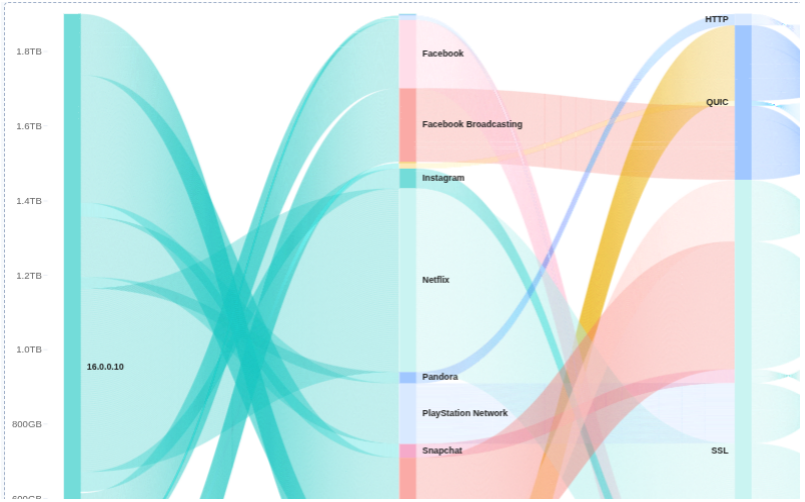
12–13 May 2026

DPDK Integration for DPI: Practical Trade-offs Between Throughput, Portability, and Operations

Harald Bunke
ipoque - a Rohde & Schwarz company

Background

- Deep Packet Inspection (DPI) enhanced Traffic Probe with DPDK packet ingress
- Live traffic monitoring and analytics (QoS, Application Detection, NW utilization)



Probe UI

Topic: Performance Impact of Setup Configuration

- Focus on high throughput
- Requires system tuning, specific NICs
- Can conflict with customer requirements & deployments

Goal:

- Reduce system setup for lower traffic volume (1-10Gbps)
- Reduce entry threshold
- Support: unspecific NICs, more deployment options

Tuning Parameter

Goal: overcome deployment challenges introduced by system tuning

Question: What performance decrease can we expect without tuning?

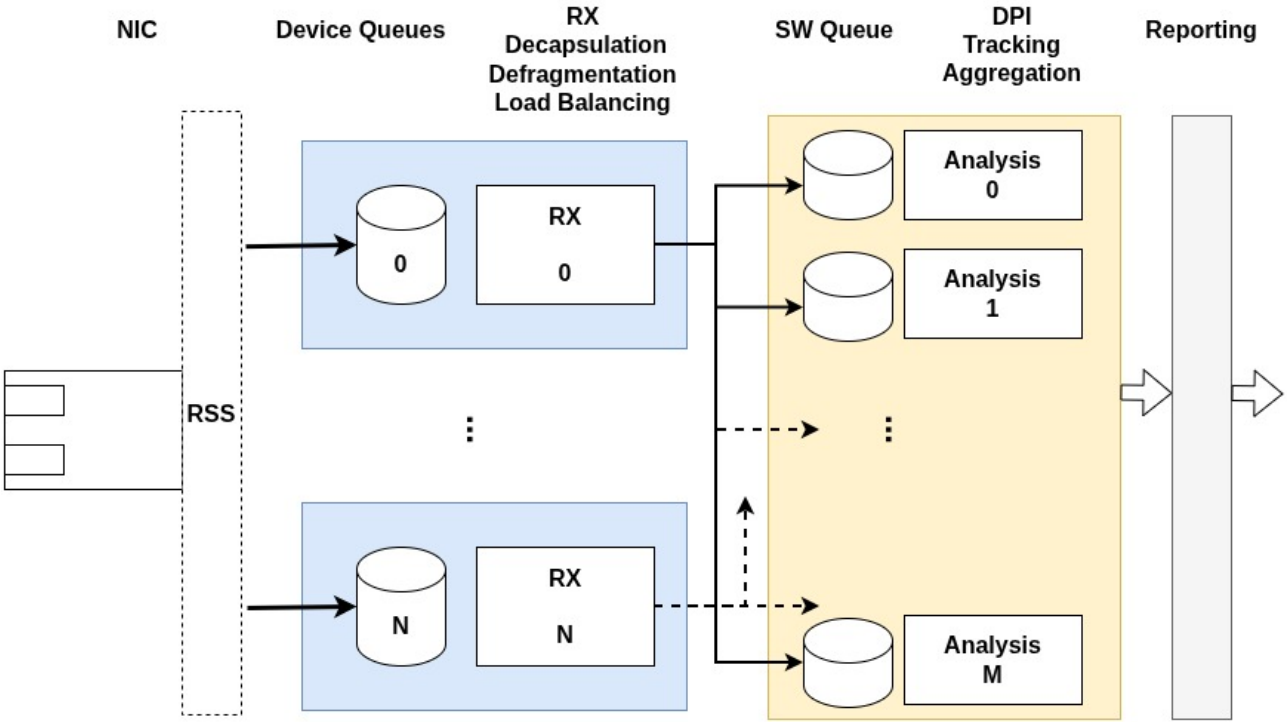
Evaluate:

- Core isolation (kernel parameter) → disable
- Hugepage allocation 1GB (kernel parameter) → do not use hugepages
- Specific NIC's → use OS driver (libpcap)

- Server HW → change compiler 'march' to 'native' → gain performance?

Design

- DPDK 24.07
- Pipeline design
- Two stages
- 1x RX core per device queue
- 1x Analysis core per SW queue



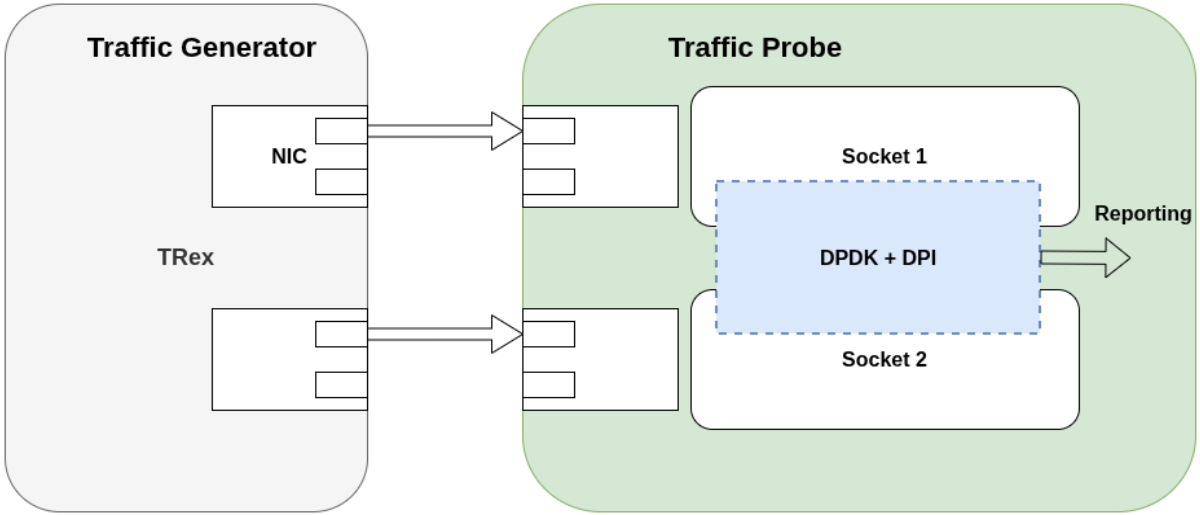
Test Setup

Traffic Probe

2x Nvidia ConnectX-6 Dx @ OFED.24.10
Dell PowerEdge R860,
Intel Xeon Platinum 8460H
(4 sockets, 40 cores/socket, disabled HT)
1024GB DRAM DDR5-4800, 256GB/Numa
Ubuntu 24.04.3 LTS

Traffic Generator

2x Nvidia ConnectX-5
2x Xeon Silver 4114
192GB DRAM DDR4-2400
Ubuntu 24.04.2 LTS
TRex v3.06

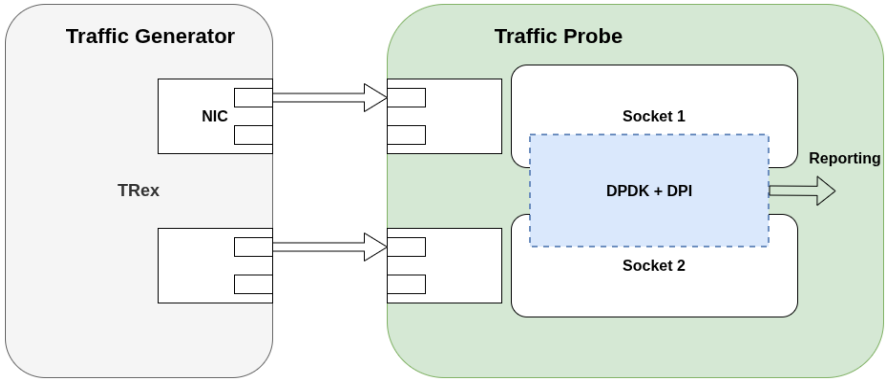


Testing

Load Test of Traffic Probe

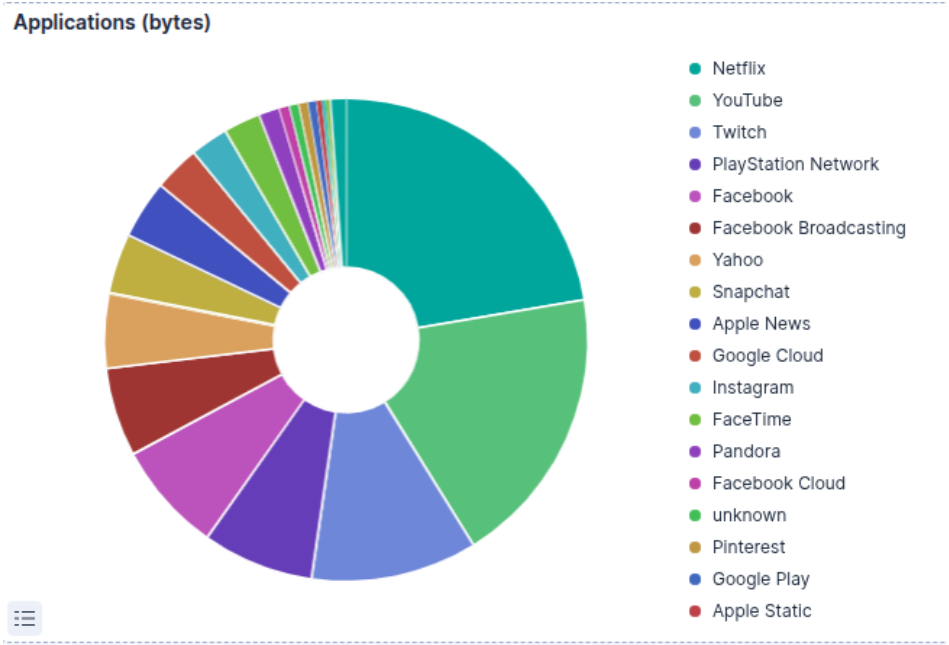
Config

- 2 NICs on 2 sockets
- 3x RX & 8x Analysis cores / NIC
- Intentionally undersized core count for max line-rate



Traffic Profile

- Different traffic load
  (up to 12 Mfps/97 Gbps) / per NIC
- Try to emulate realistic traffic
- TRex Profile with 48 PCAPs
- MTU: 1500 bytes
- Load test: 20 minutes

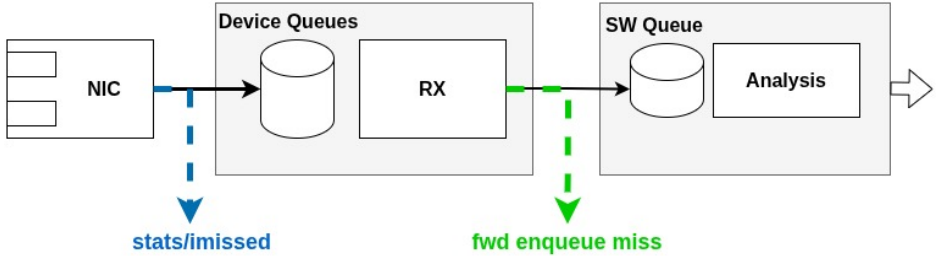


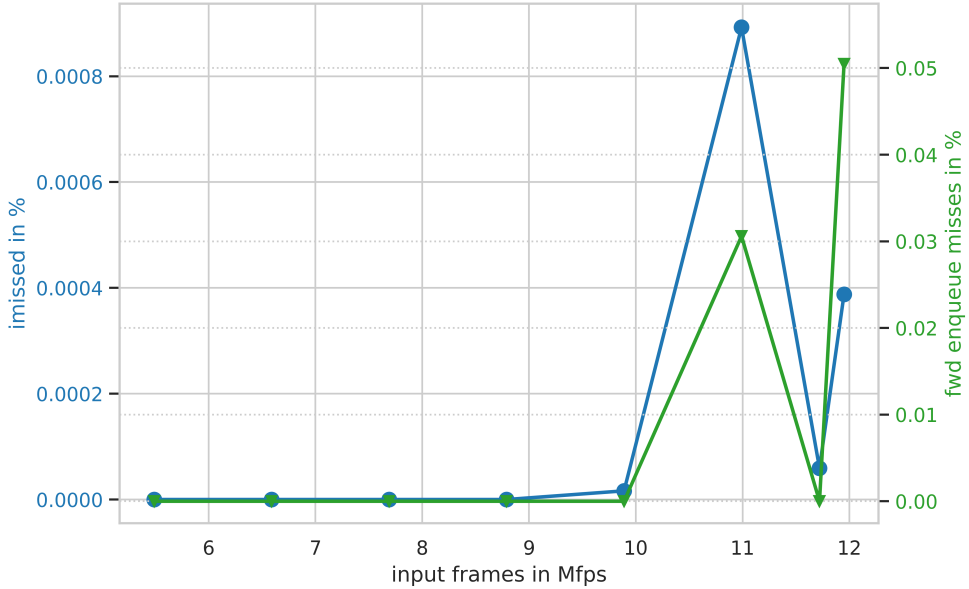
Normal

- Driver: mlx5, isolated cores, 1GB hugepages, march: corei7

- Average of both NIC's
- Blue: imissed in %
- Green: enqueue failures for SW queue in %

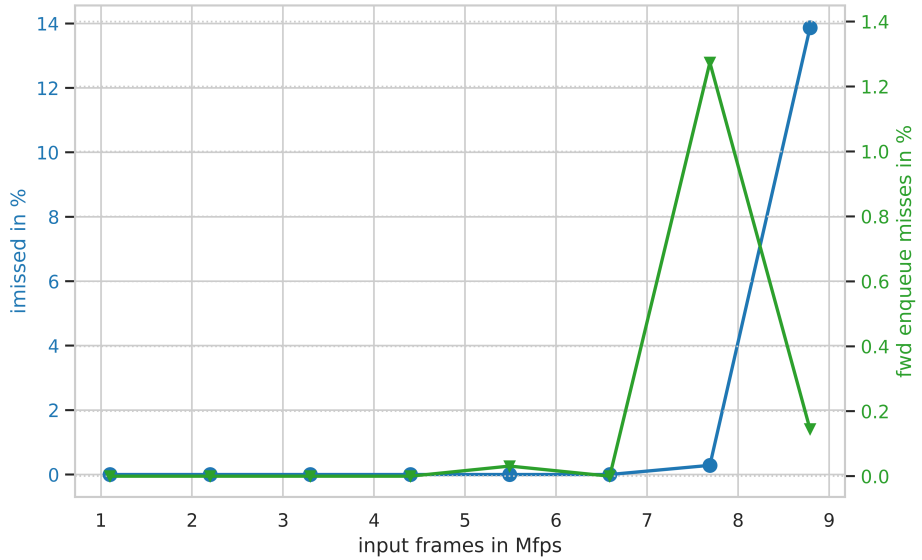
- Drops at > 8.8 Mfps, low drop rates
- Expected due to configuration
- Missing frames → can disrupt analysis





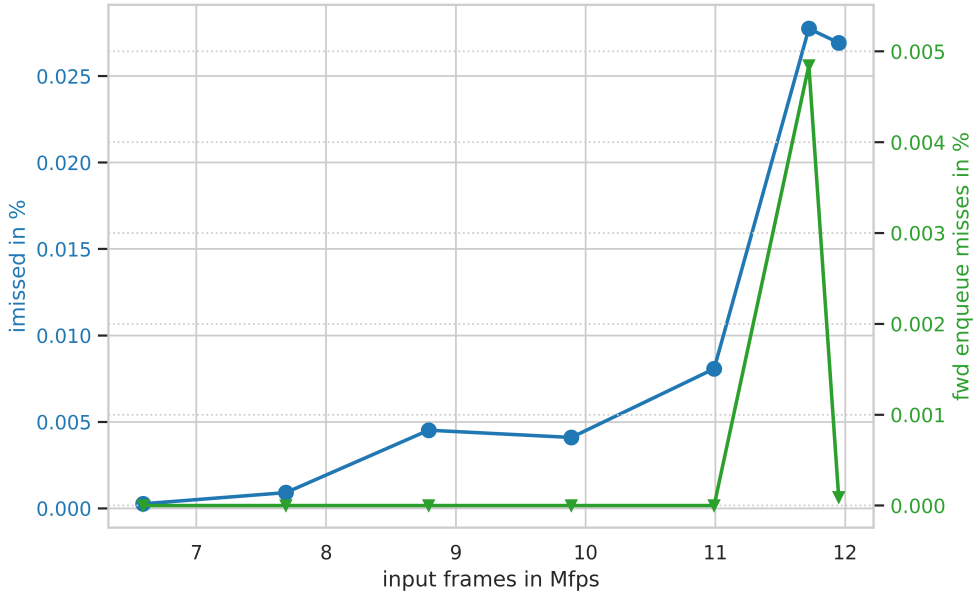
No Hugepages

- EAL options: '--no-huge', '-m <memory>'
- Hugepages only used for mbuf
- Drops at > 4.4 Mfps
- Then high drop rates



No CPU Isolation

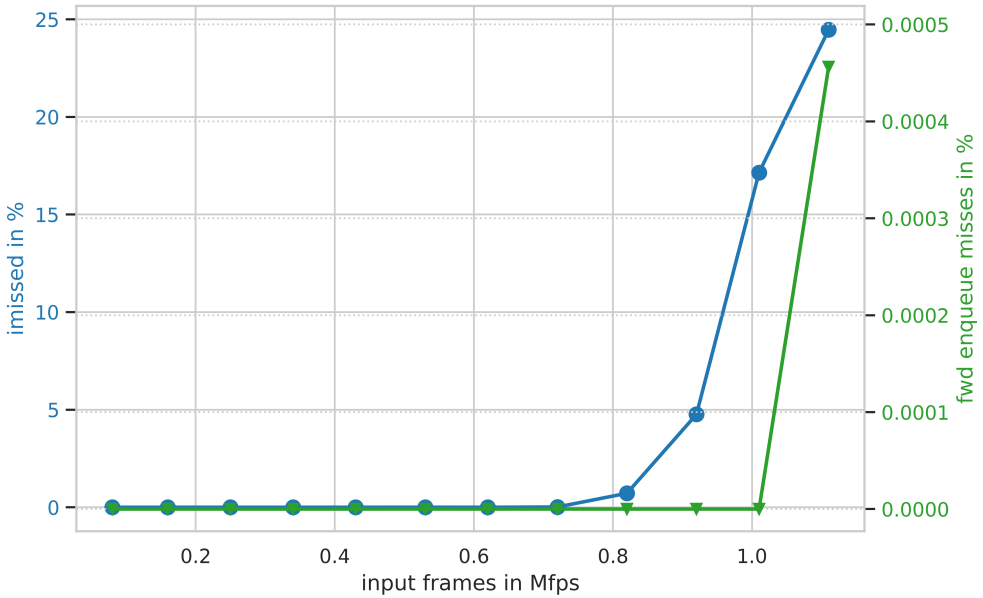
- Used same cores
- Scheduled other tasks on DPDK cores
- Drops at > 6.6 Mfps
- Low drop rates



PMD libpcap

- Via virtual device
- Only one device queue
- 1x RX & 8x Analysis core

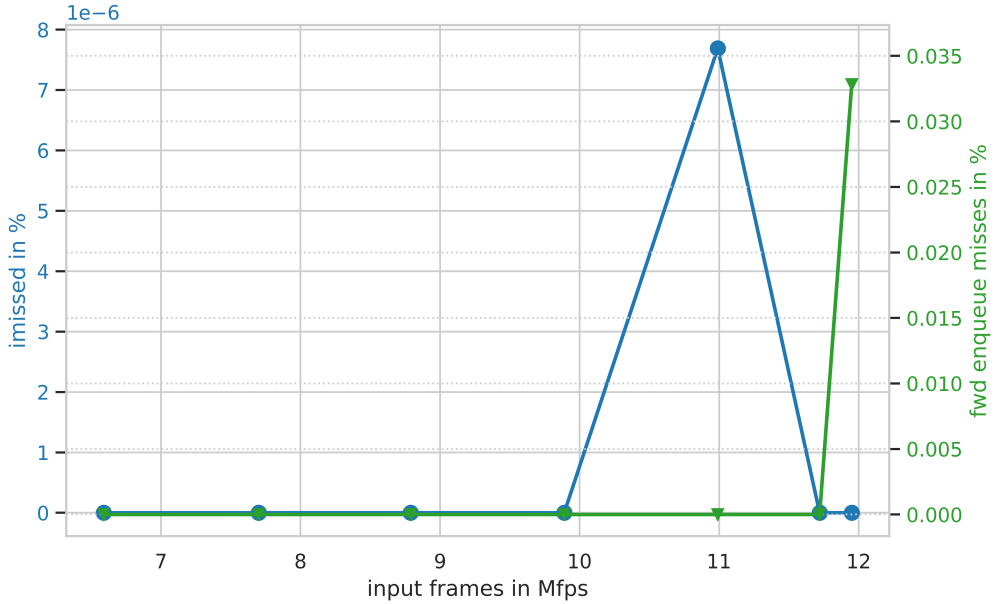
- Occasional imissed already at 0.16 Mfps (5‰) → burst in profile volume
- Maybe up to 0.72 Mfps
- Maybe tweaks for driver/integration



Native Build

- Identified a previously unevaluated performance tweak
- Changing '-march=corei7' to '-march=native' for DPDK build
- gcc 11.4.0
- Not portable, customer specific

- Drops at > 9.8 Mfps
- Performance gain



Summary

- Straight forward integration
- Some open tasks/questions for better integration

Outlook:

- Test runtime hugepages
- Combined settings
- Try other PMD
- Evaluate other features/libraries form DPDK

Config	Zero drop input in Mfps	Zero drop input in Gbps	Performance Factor (normal as reference)
libpcap	0.16 (0.72)	1.28 (5.76)	0.02 (0.08)
No Hugepages	4.4	35.2	0.5
No CPU isolation	6.6	52.8	0.75
Normal	8.8	70.4	1
Native march	9.8	78.4	1.11

DPDK SUMMIT
Powering the Future of Networking Software

12–13 May 2026

Questions?

Harald Bunke
ipoque - a Rohde & Schwarz company

https://www.ipoque.com


DPDK
SUMMIT
Powering the Future of Networking Software