iWave Global

Programmable Data Planes for the Cloud Era

- Harnessing P4 on FPGA based SmartNICs

A global leader in high performance FPGA & ODM solutions, backed by engineering excellence

1999
26 Years of Expertise

320+
Talented R&D Team

Strategic Industry Focus
Aerospace and Defense
Industrial
Medical
Automotive
Networking

Bangalore, India

Ras Al Khaimah, UAE

California, USA

New Jersey, USA

Ratingen, Germany

Gyeonggi-do, Korea

Taipei City, Taiwan

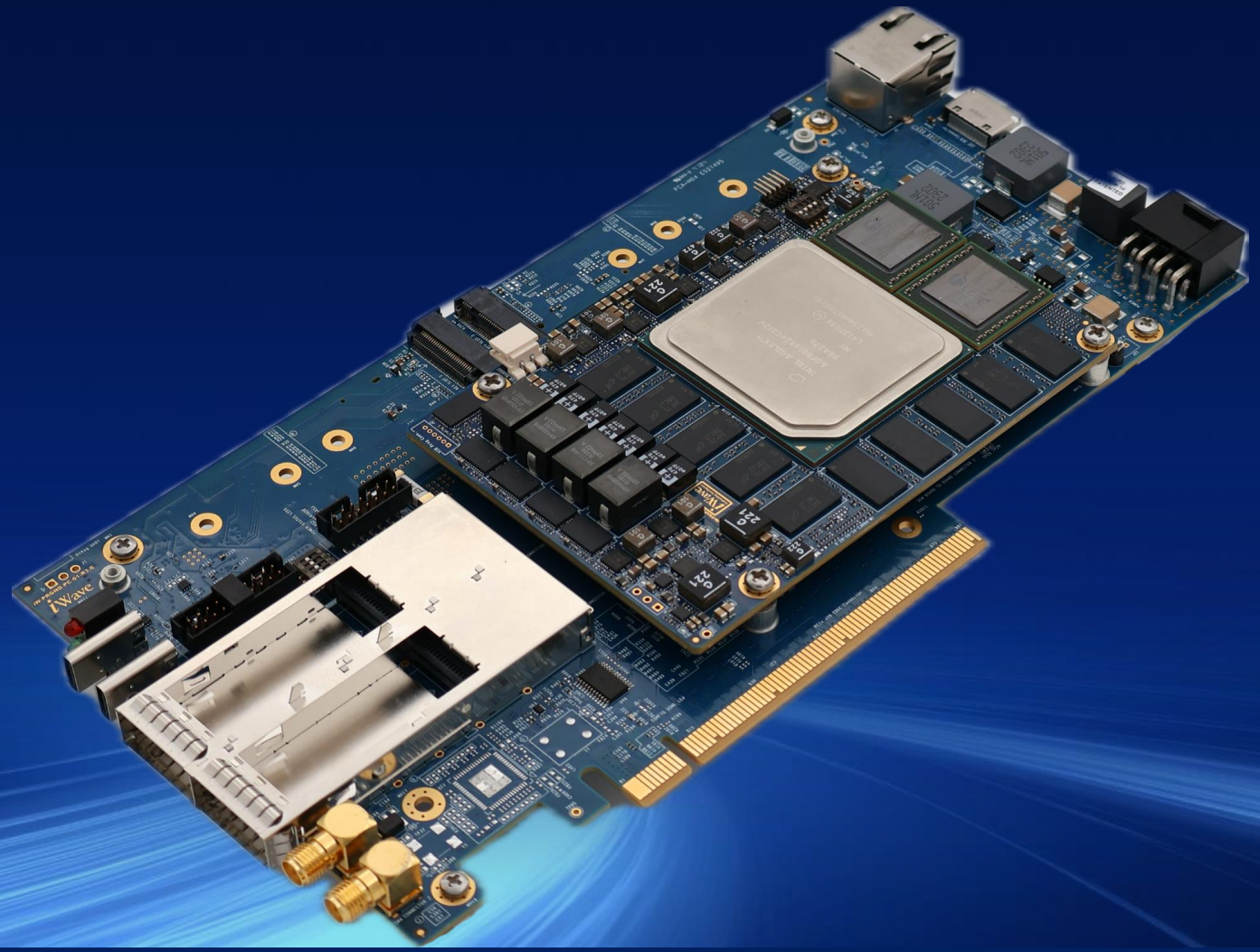
Yokohama, Japan

iWave Global

mktg@iwave-global.com

Agenda

- Motivation & Need for a SmartNIC
- Why FPGA SmartNICs?
- Application Offloading
- Packet Processing & Shell Framework
- P4 Compilation Flow
- Workloads/Applications – use cases.
- iWave Offerings
- Live Demo



Motivation & Need for a SmartNIC

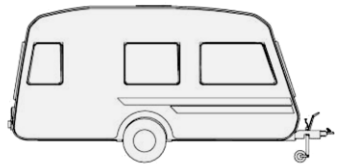
SmartNIC

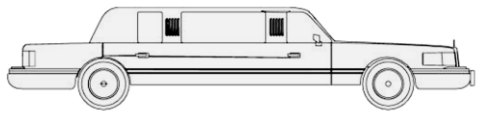
?

Why Packet Processing Needs to Change

- Explosive growth of:
  - Cloud-native workloads
  - Overlay networks (VXLAN, IP-in-IP)
  - 5G and data-intensive applications
- Increasing need for:
  - Custom tunnelling
  - Dynamic QoS
  - Fast feature rollout

Legacy Networking	Modern Networking
• Fixed Protocols • ASIC Pipelines • Slow Change	• Overlays • Custom Headers • Fast Change





The network is changing faster than the hardware.

The challenge today isn't processing packets fast—it's adapting to new packets without waiting for new silicon.

Why FPGA in SmartNICs?

- Line-rate packet processing
- Reconfigurable Pipelines
- Isolates & Offloads host CPU
- Ideal Target for P4.
- Removes low-level FPGA coding complexity
- Bridges gap between:
  - Software flexibility
  - ASIC-level performance

Conventional Approach

- HDL or HLS based coding
- RTL Expertise
- Rerun FPGA Process – RTL coding, Syn, PnR
- Long Development Cycles

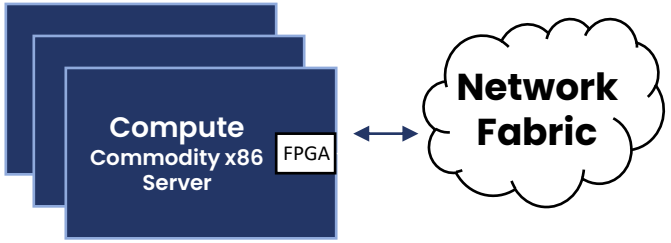
P4 Approach

- Domain Specific Protocol
- Protocol Independence & Cross – Platform Portability
- Rapid Development
- Field Reconfigurability
- Higher Abstraction Layer

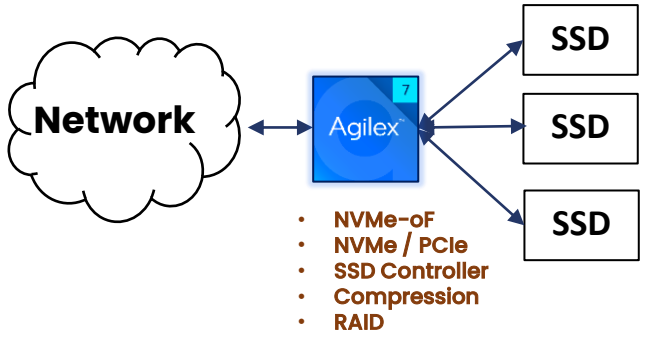
FPGA Programming !!!

SmartNIC Applications

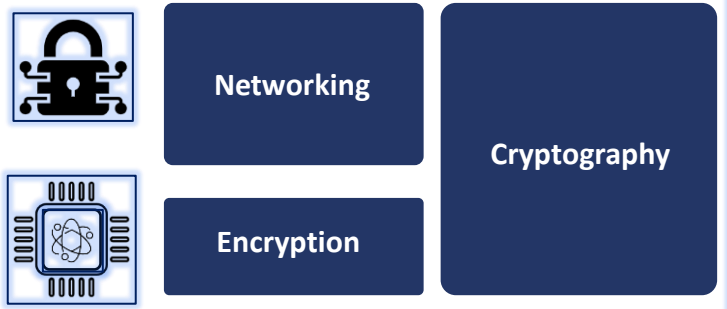
HPC & Networking

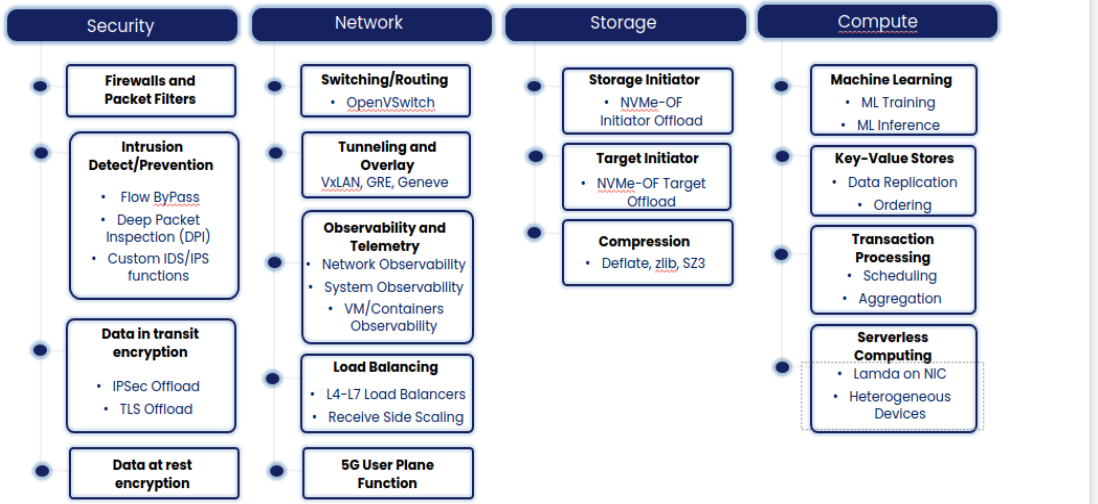


Storage Appliances

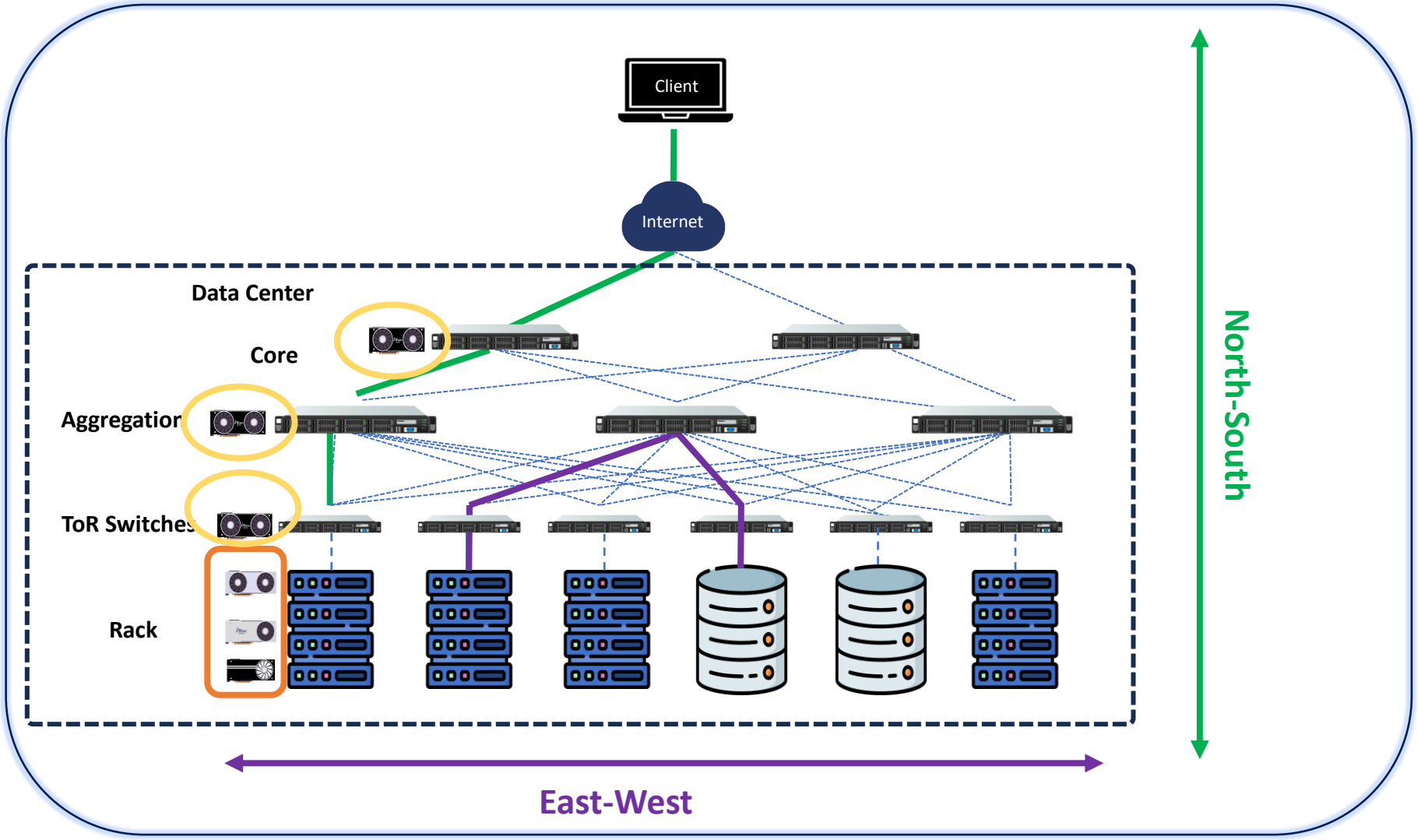


Security





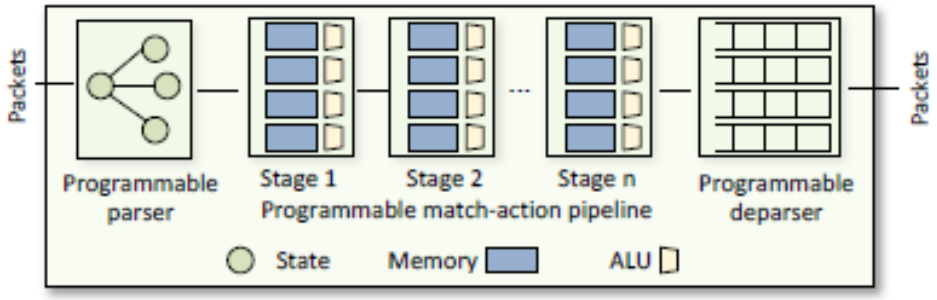
Security

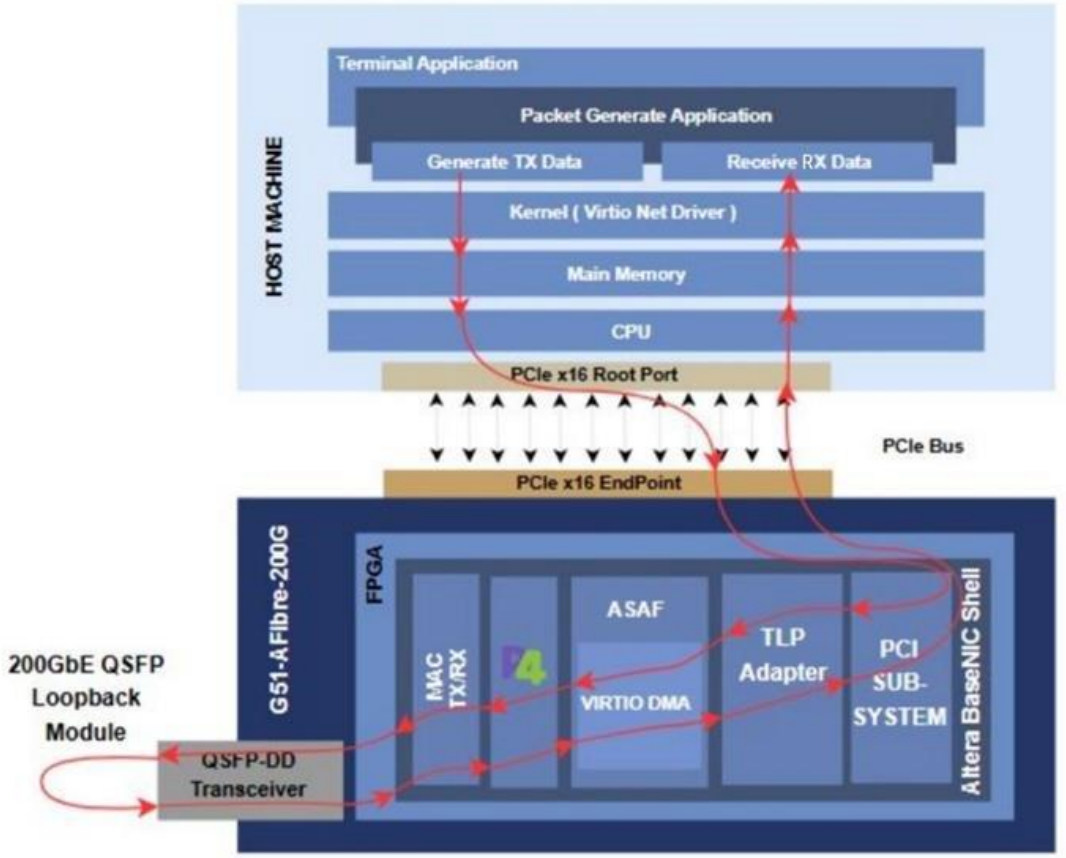

Client
Internet
Data Center
Core
Aggregation
ToR Switches
Rack
North-South
East-West

Architecture Overview

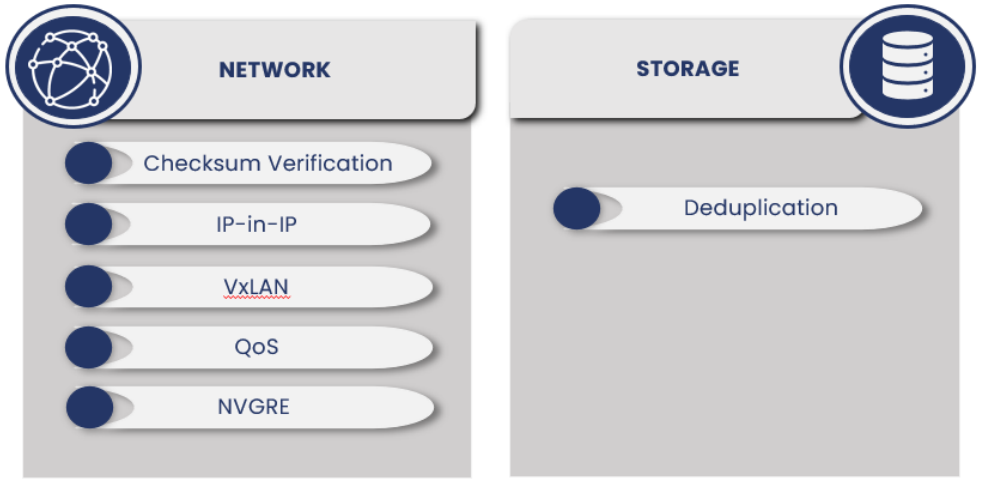
P4 on FPGA iW-Fibre SmartNIC: End-to-End View

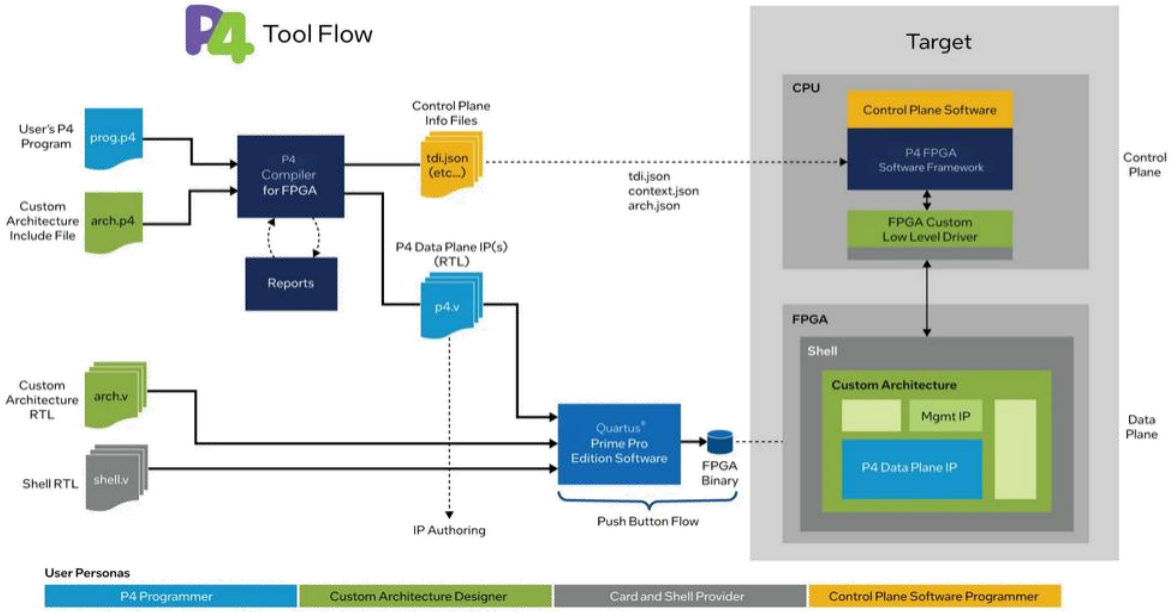
- Data plane – Control Plane - Management Plane
- Acceleration:
  - Host Software
  - Hardware

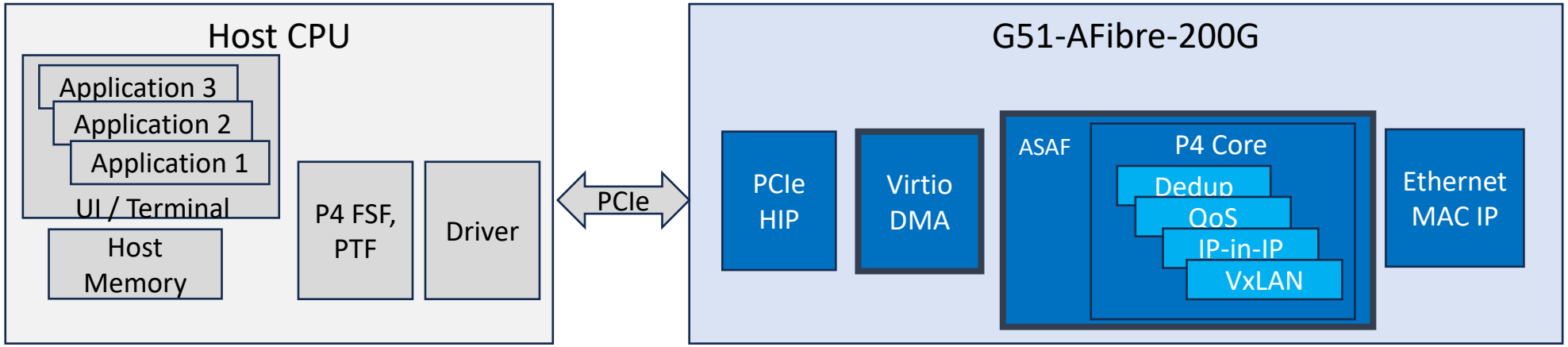




Demo: P4 based Programmable Packet Processing for Custom Workloads







Altera Quartus IP

Altera Proprietary IP

iWave

Use Case 1: Checksum Verification

Problems

- Virtual machines (VMs) or containers send/receive packets at high speed
- Packet corruption or misrouting occurs due to network errors, faulty switches, or misconfigured virtual NICs
- Debugging at scale is time-consuming and impacts SLA compliance

Where CKSUM Used?

- Packet integrity & error detection
- TCP/UDP/IP header validation
- Tunneling protocols (VXLAN, GRE, GENEVE)
- Offloaded data center networking & cloud acceleration

CKSUM on SmartNIC

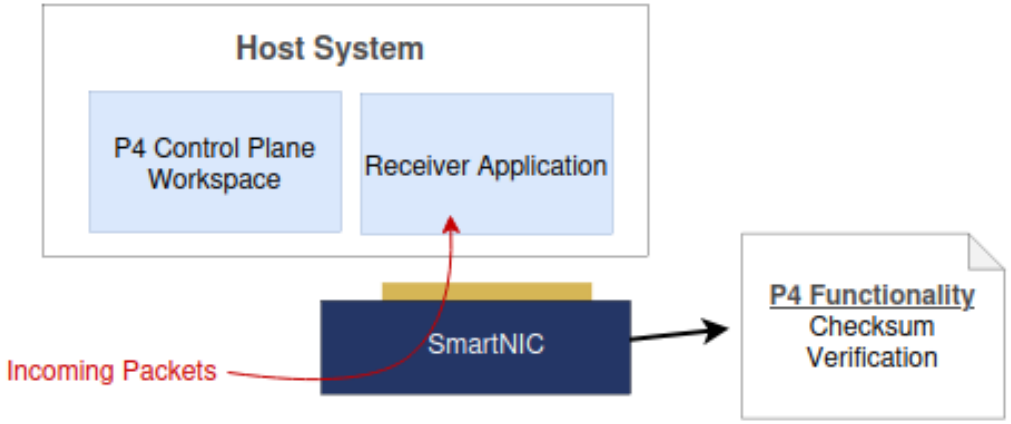
- Track per-flow checksum errors in real time
- Rapidly detect corrupted packets without burdening host CPU
- Allows operators to proactively debug, isolate faulty paths, and maintain service reliability

Demo 1: Checksum Verification

Objective

- Packet decapsulation at line rate.
- Verify packet and Forward only valid packets to host.
- Drop or mark corrupted packets before reaching host
- Reduce host CPU usage through offload

Data Flow



For More Details

P4 on iW-Fibre SmartNICs Packet Forwarding and Checksum Offload

Use Case 2: IP-in-IP

Problems

- Data center or cloud network handling millions of encapsulated IP-in-IP packets
- Packet corruption or misrouting can occur during encapsulation/decapsulation
- Verifying packet integrity at high speed on the host CPU creates bottlenecks

Where IP-in-IP Used?

- Data center network overlays – tunneling tenant traffic across the physical network
- Multi-tenant cloud environments

IP-in-IP SmartNIC

- Offloads encapsulation and decapsulation from host CPU
- Performs per-packet checksum verification in real time
- Reduces CPU load and improves throughput
- Enables line-rate packet processing and early detection of corrupted packets

Use Case 3: VxLAN

Problems

- Debugging VXLAN tunnel issues is slow
- Multi-tenant overlays increase complexity for network troubleshooting
- Host CPU becomes a bottleneck for parsing and telemetry

Where VxLAN Used?

- Virtualizes Layer-2 domains over Layer-3 IP fabric
- Interconnects thousands of VMs and containers across racks
- Enables scalable multi-tenant networks using overlays
- Uses VTEPs (VXLAN Tunnel Endpoints) to encapsulate and decapsulate packets

VxLAN on SmartNIC

- Offloads VXLAN parsing, encapsulation, and decapsulation to hardware
- Reduces host CPU usage and improves throughput
- Enables line-rate processing for multi-tenant overlays

Use Case 4: NVGRE

Problems

- High-speed multi-tenant traffic can cause packet misrouting or corruption
- Troubleshooting overlay network issues is slow and complex
- Host CPU struggles with real-time encapsulation, decapsulation, and integrity checks
- Maintaining per-tenant visibility and SLA compliance is difficult

Where NVGRE Used?

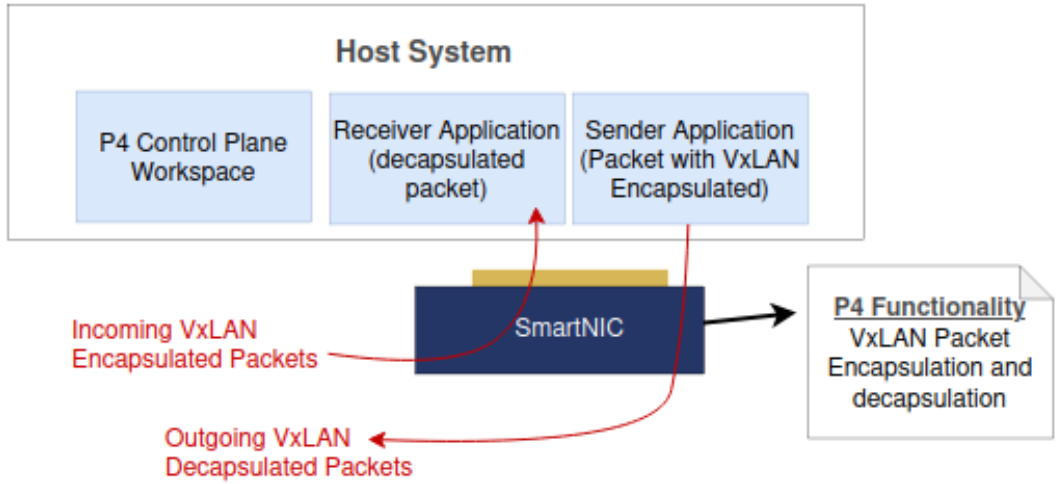
- Data center network overlays – tunneling tenant traffic across the physical network
- Multi-tenant cloud environments

NVGRE SmartNIC

- Offloads encapsulation and decapsulation from host CPU
- Performs per-packet checksum verification in real time
- Reduces CPU load and improves throughput
- Enables line-rate packet processing and early detection of corrupted packets

Demo : Tunneling

Data Flow



Use Case 5: QoS

Problems

- Different types of network traffic have different requirements (delay-sensitive, throughput-sensitive, best-effort)
- All packets treated equally without QoS can delay critical services (e.g., video calls)
- Multi-tenant VXLAN overlays increase complexity in traffic management
- Host CPU is burdened with parsing, classification, and rate-limiting at high speed

Where CKSUM Used?

- Ensures priority for critical applications (e.g., VoIP, video conferencing)
- Guarantees throughput for bulk transfers (file backups, system updates)
- Supports multi-tenant traffic isolation and fairness

VxLAN on SmartNIC

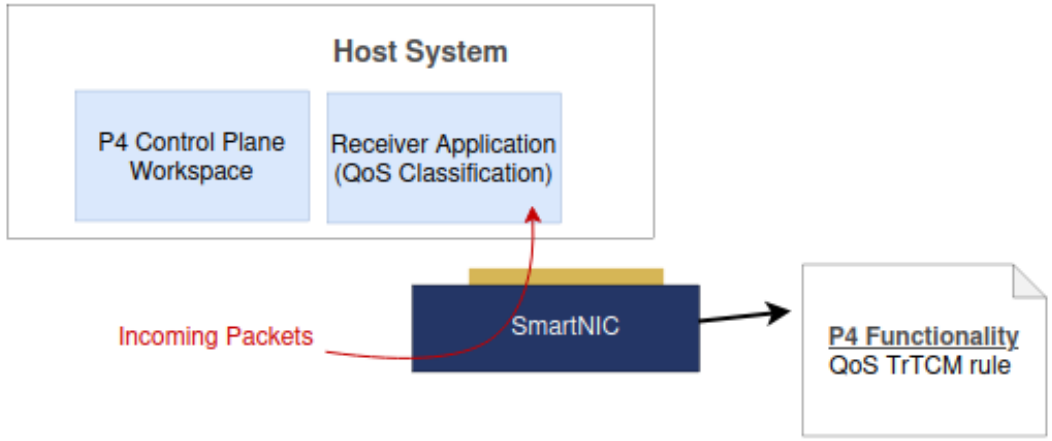
- Offloads VXLAN parsing, encapsulation, and decapsulation to hardware
- Reduces host CPU usage and improves throughput
- Enables line-rate processing for multi-tenant overlays

Demo 5: QoS

Objective

- Classify packets into QoS classes (Voice, Video, Best-Effort)
- Apply rate-limiting and marking in hardware
- Show reduced host CPU usage and scalable traffic management

Data Flow



For More Details

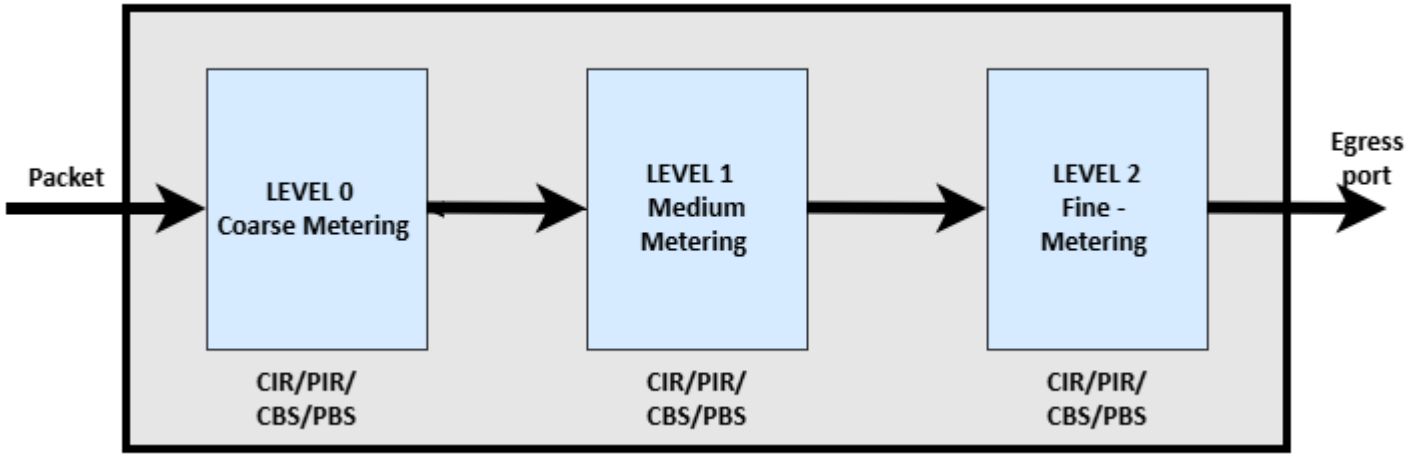
P4-Based Packet De-duplication on iW-Fibre SmartNICs

Demo 5: QoS

Packet Classification

Protocol Type	Traffic Characteristics	DSCP Value	DiffServ Class	QoS Intent
TCP	Reliable, throughput-oriented traffic	46	Expedited Forwarding	High priority, low latency
UDP	Latency-sensitive traffic (e.g., voice/video)	44	Voice Admit	Voice Admit
Other protocols	Default traffic	0	Best Effort	TStandard forwarding

Rate Limiting



Use Case 6: Data Deduplication

Problems

- Bandwidth Waste : Duplicate packets or data segments consume unnecessary bandwidth.
- Storage Overhead : In storage networks, sending/storing duplicate data increases storage requirements.

Where are Deduplication Used?

- WAN optimization : Deduplication reduces the amount of data that needs to be transmitted over long distances, which saves bandwidth and accelerates application performance.
- Storage systems and backups

Deduplication in SmartNIC

- Reduced bandwidth and latency : By eliminating redundant packets, it reduces the amount of data transferred over the PCIe bus and processed by the host.
- Increased storage efficiency: Deduplication reduces the amount of physical storage required for data volumes by storing only unique data blocks.

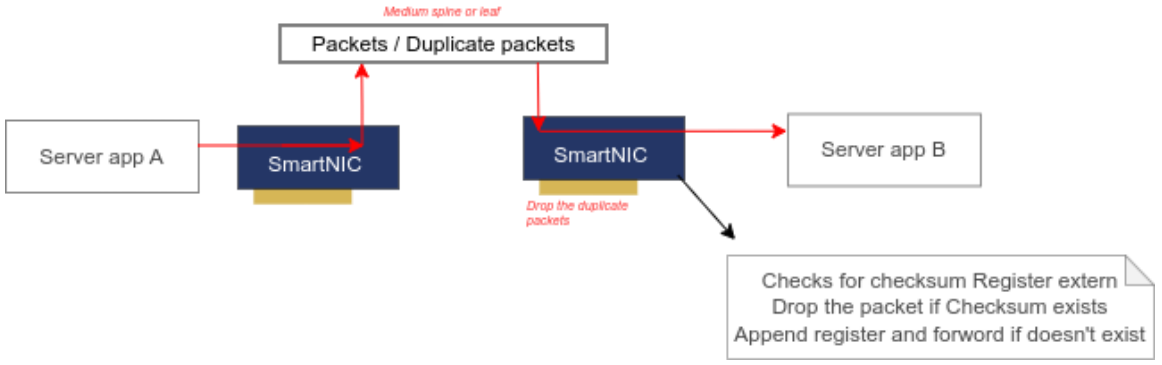
Demo 6: Data Deduplication

Objective

- Show dynamically dropping redundant traffic at line rate
- Demonstrate the throughput of host CPU as we filtered the repeated packets
- Visualize **live metrics** : latency, throughput and integrity

Setup 2

Data Center View

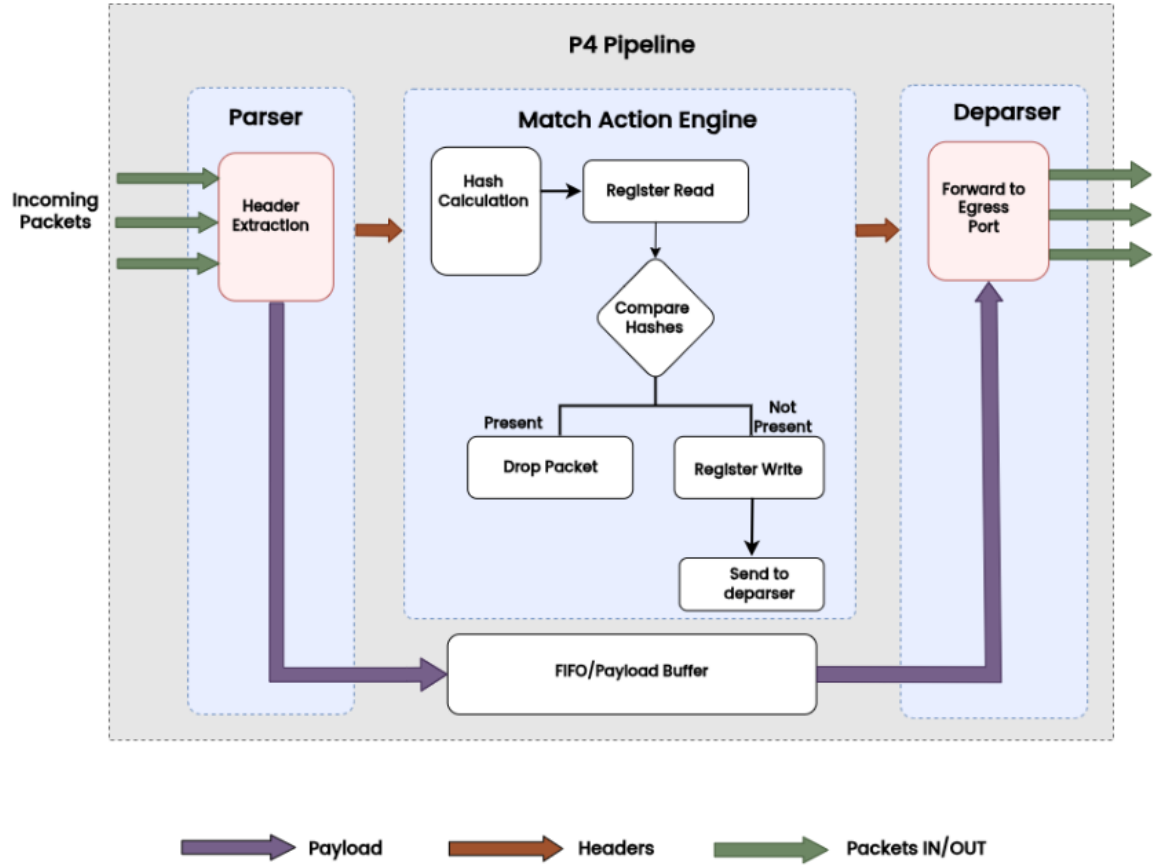


For More Details

P4-Based Packet De-duplication on iW-Fibre SmartNICs

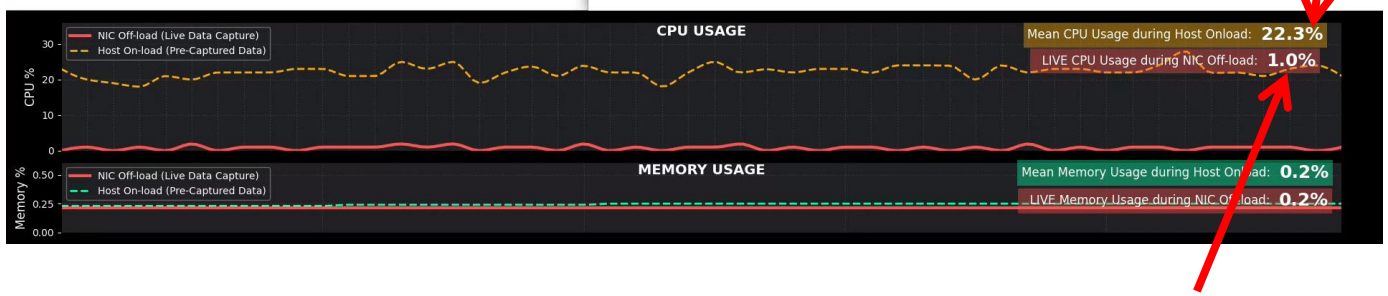
Demo 6: Data Deduplication

Packet De-duplication on the iW-Fibre SmartNIC



Performance Observations & Key Takeaways

Host On-load stat



iW-NIC Off-load stat

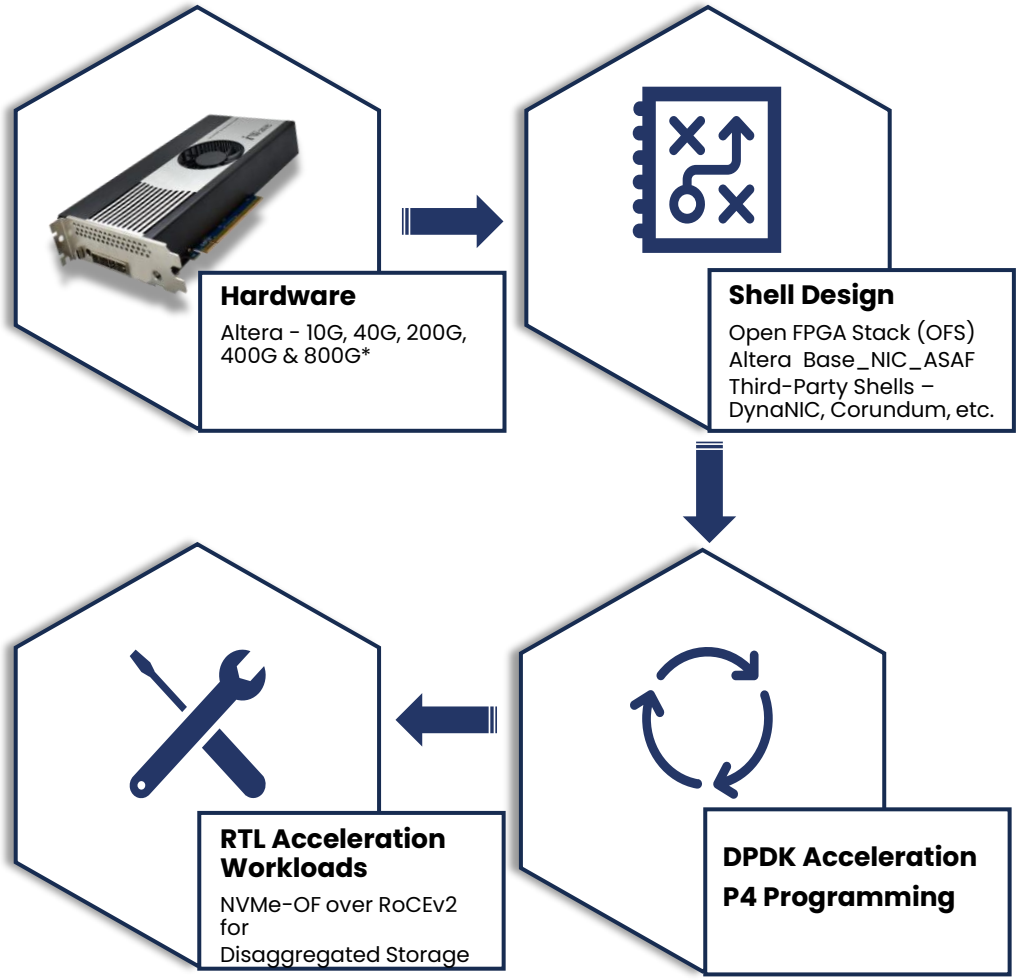
Test Environment	
PC	Dell Precision 3680
SmartNIC	iW-Fibre SmartNIC
Standard NIC	Mellanox ConnectX-5 (Without Offload enabled)
CPU Cores	20
Operating System	Ubuntu 22.04 LTS (Kernel 6.8.0-85-generic)
Memory	64 GB DDR5 RAM
Driver	Virtio-Net
Host PCIe Spec	Gen5

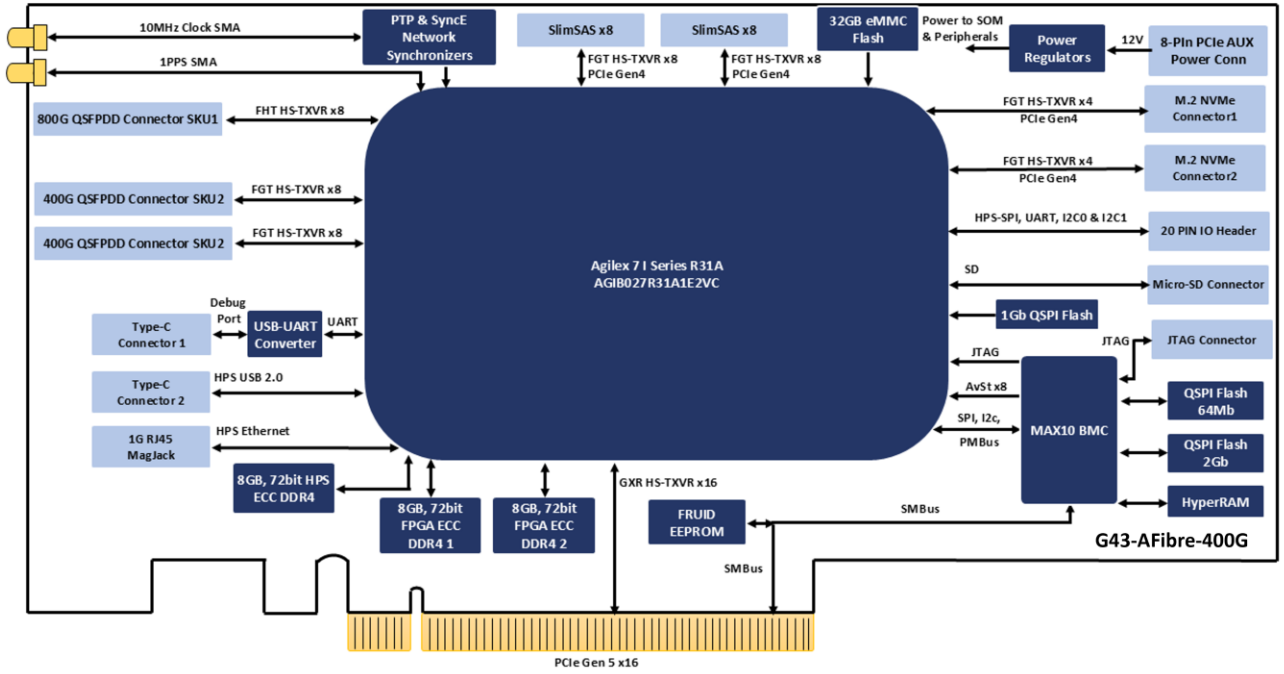
P4 Development	
Development Tool	Quartus Tool, Altera P4 Suite and P4 Compiler
Test Environment	FPGA Software Framework(FSF), Packet Test Framework (PTF)
Packet Capture	Scapy, Tcpdump, libpcap

Resource Utilization		
Network Workloads	**NIC Off-load CPU (%)**	**Host On-load CPU (%)**
Checksum	1.0	22.3
IP in IP	1.9	16.6
VxLAN	1.9	30.2
QoS	1.9	13.4
NVGRE	1.9	30.6
Deduplication	1.9	22.8

Demo Link : https://youtu.be/J55-0qT1lJA

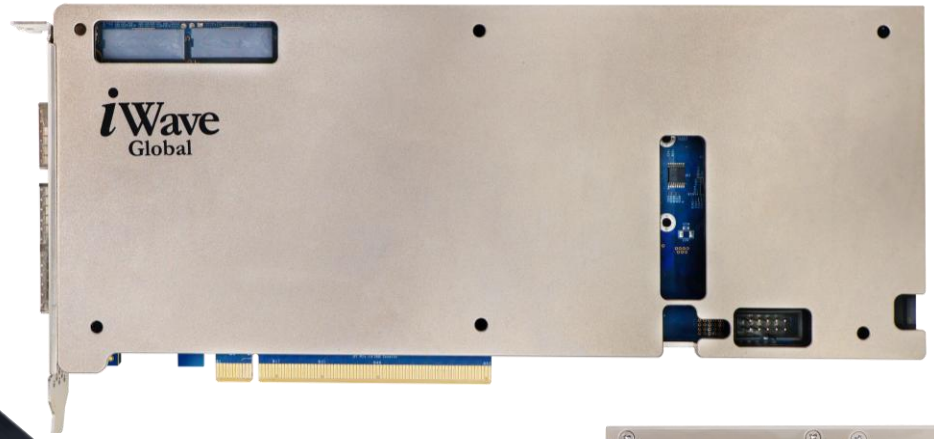
Demo to Deployment

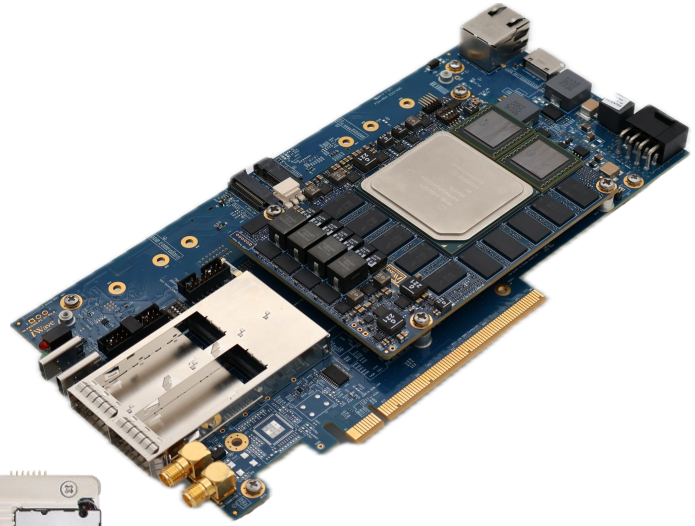


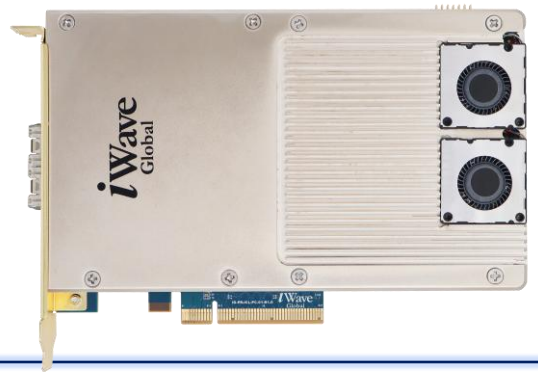


iWave iW-Fibre SmartNIC Offerings

1. **Agile Customization:**
2. **Full-Stack Design Authority**:
3. **Ecosystem Collaboration**:





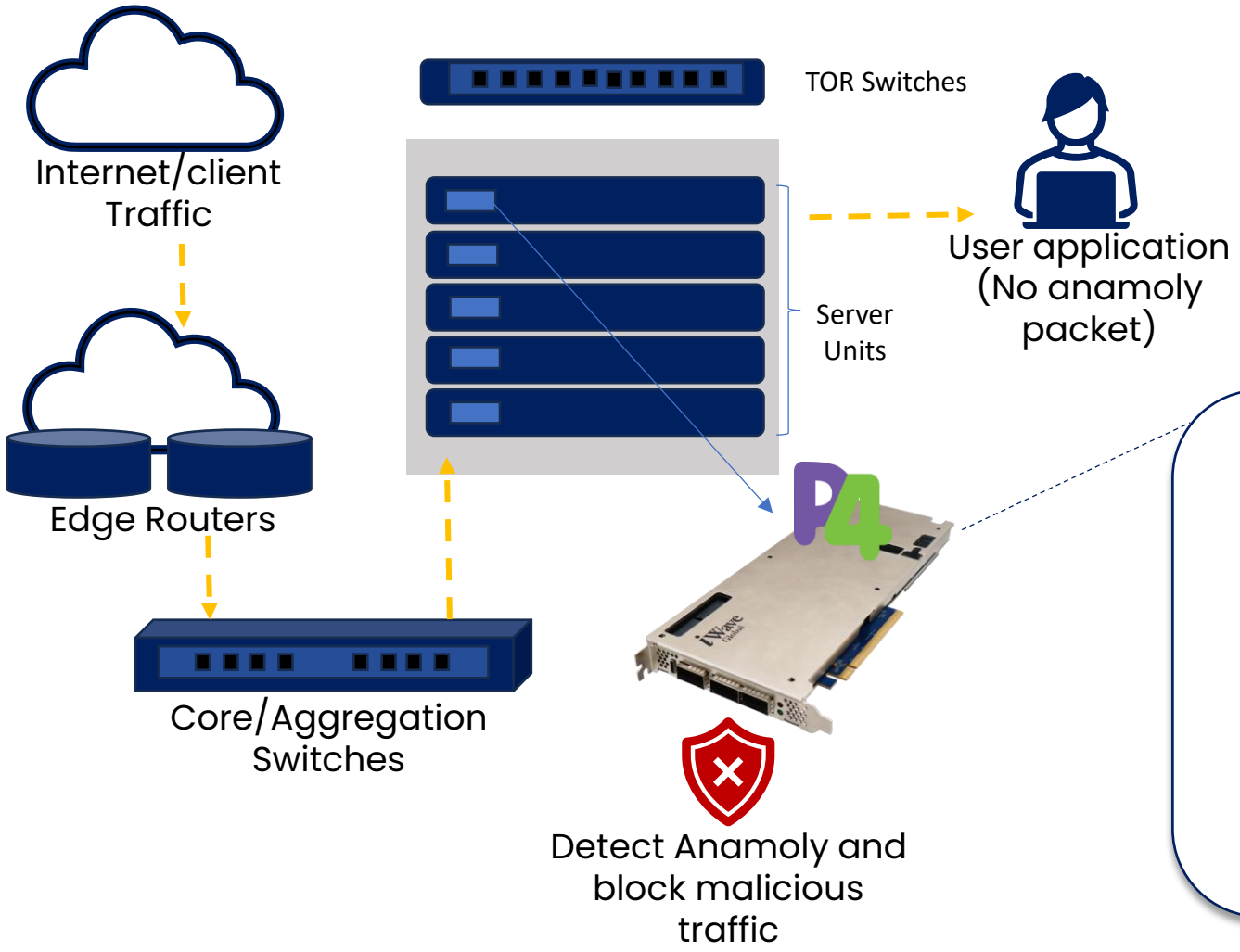


Live Demo, References & Q&A

References

- Altera P4 Suite for FPGA : https://www.altera.com/products/development-tools/p4-suite-fpga
- Altera BASE_NIC & ASAF Framework.
- A Comprehensive Survey on SmartNICs: Architectures, Development Models, Applications, and Research Directions

Network Intrusion Prevention



- Inspects packets at **line rate**
- Performs **flow tracking, protocol parsing, and intrusion prevention**
- **Malicious traffic is detected and blocked immediately**
- Only **clean, policy-compliant traffic** is forwarded upward

P4 based Network Intrusion Prevention

P4 Main parser

main.p4

```
Main_Pipeline_0 (
    Main_0_Parser(),
    Main_MA_0(),
    Main_Deparser())
pipe_main_0;
```

tables.p4

```
//Match action
  if (hdr.tcp.isValid()) {
    // Verify the dst ports
      if (http_ports.apply().hit) {
      // Verify signature
          if (hdr.sign.isValid()) {
    if (hdr.sign.vul1 == 0x74657874) // text
      if (hdr.sign.vul2 == 0x000000)
        if (hdr.sign.vul3 == 0x45256D){
            drop(); //drop the packets
                }
              }
        }
  }
```

parser_main.p4

```
    state parse_ipv4 {
        pkt.extract(hdr.ipv4);
        dscp = hdr.ipv4.dscp;
        verify(hdr.ipv4.version == 4w4, error.IPv4IncorrectVersion);
md.ipv4_hdr_len = hdr.ipv4.minSizeInBytes();
        transition select(hdr.ipv4.protocol) {
            L3_PROTO_TCP      : parse_tcp;
            L3_PROTO_UDP      : parse_udp;
            default           : accept;
        }
    }

    state parse_tcp {
pkt.extract(hdr.tcp);
md.tcp_hdr_len = hdr.tcp.minSizeInBytes();
transition parse_sign ;
    }

    state parse_udp {
      pkt.extract(hdr.udp);
      md.udp_hdr_len = hdr.udp.minSizeInBytes();
transition parse_sign ;
    }

    state parse_sign {
      pkt.extract(hdr.sign);
transition accept ;
    }

```

header.p4

```
header ipv4_h {
    bit<4>  version;
    bit<4>  ihl;
    bit<6>  dscp;
    bit<2>  ecn;
    bit<16> totalLen;
    bit<16> identification;
    bit<3>  flags;
    bit<13> fragOffset;
    bit<8>  ttl;
    bit<8>  protocol;
    bit<16> chksum;
    bit<32> srcAddr;
    bit<32> dstAddr;
}
header udp_h {
    bit<16> srcPort;
    bit<16> dstPort;
    bit<16> len;
    bit<16> checksum;
}
header tcp_h {
    bit<16> srcPort;
    bit<16> dstPort;
    bit<32> seq;
    bit<32> ack;
    bit<4> dataOffset;
    bit<3> res;
    bit<9> flags;
    bit<16> window;
    bit<16> checksum;
    bit<16> urgentPtr;
}
header signature_h {
    bit<624> pad1;  // 78 bytes
    bit<32> vul1;   // 4 bytes (text)
    bit<8> pad2;    // 1 byte
    bit<24> vul2;   // 3 bytes (0x000000)
    bit<8> pad3;    // 1 byte
    bit<24> vul3;   // 3 bytes (0x45256D)
}
.
.
.

```

Thank You

We are here to
Accelerate Embedded Innovation

mktg@iwave-global.com

INDIA | USA | GERMANY | EUROPE | TAIWAN | KOREA | JAPAN | DUBAI