Network Acceleration XDF 2018

Ravi Sunkavalli
VP, DC & IP Design Solutions

XILINX.

© Copyright 2018 Xilinx

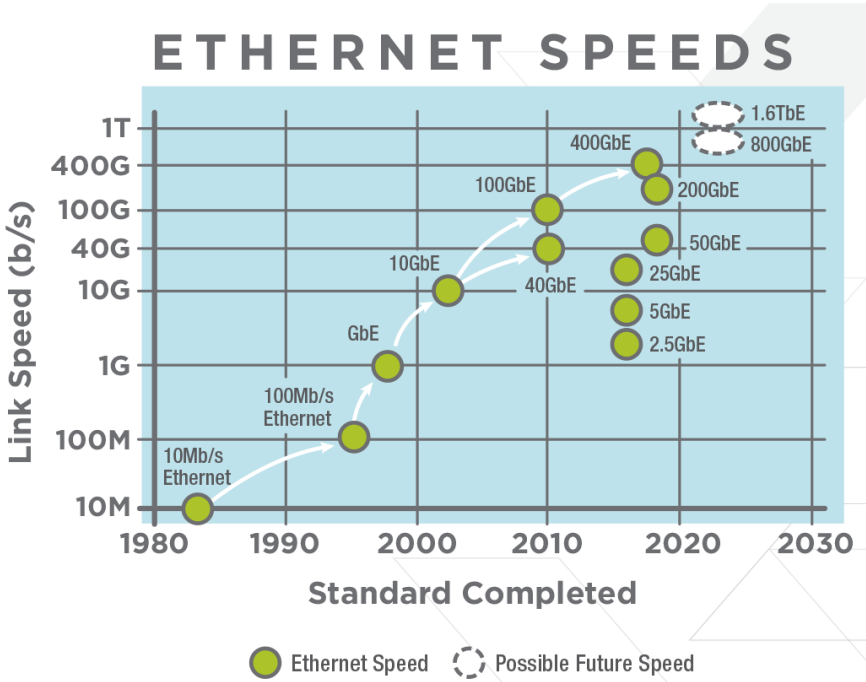
Agenda

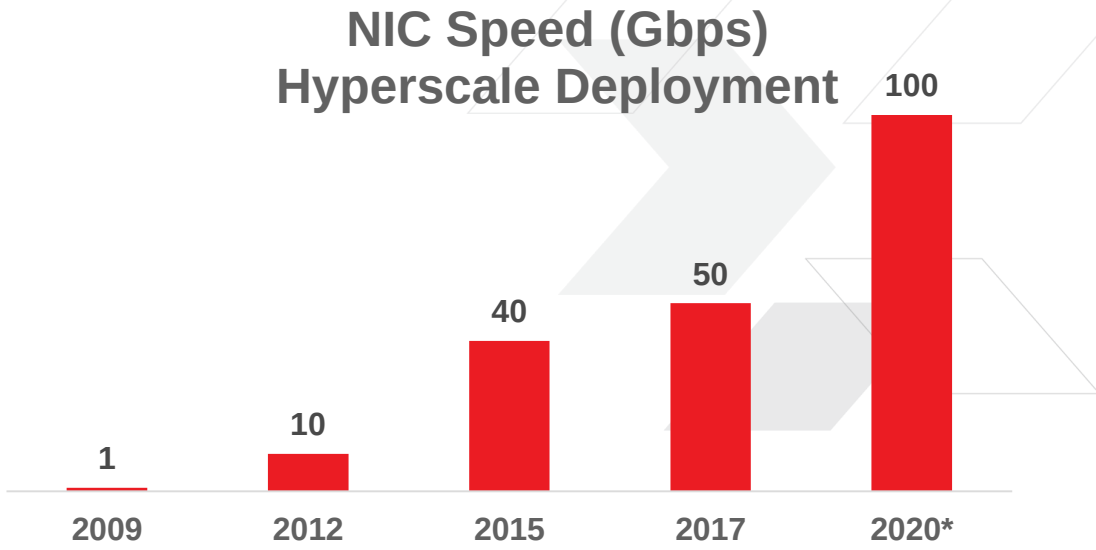
- Networking Industry Trends
- Network Acceleration Platform & Usecases
- Key enabling IP, Tools & Technology

© Copyright 2018 Xilinx

Server Networking Trends

- Data rates increasing rapidly
- CPU Performance not scaling
- Virtualization & Bare Metal
- High Feature Velocity

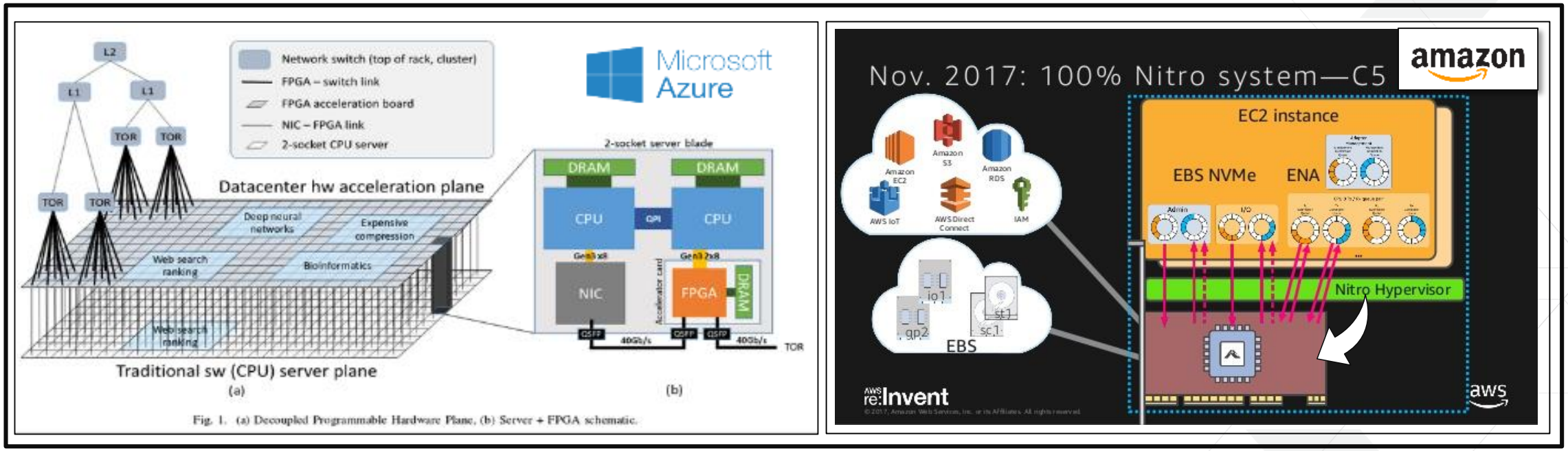




© Copyright 2018 Xilinx

XILINX.

SmartNICs in Hyperscale Cloud

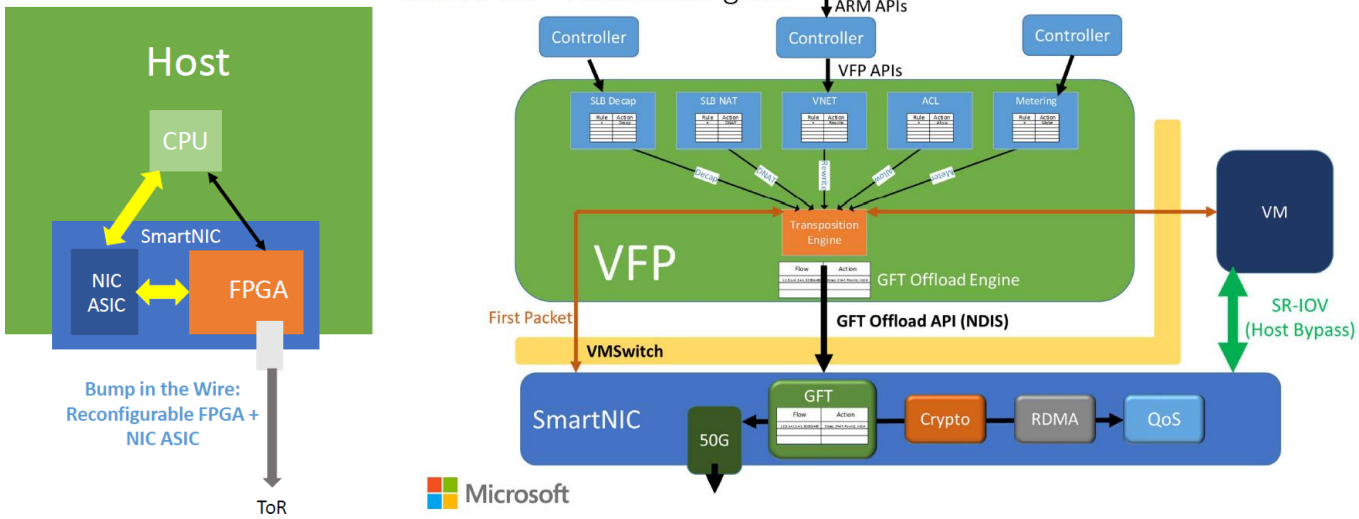


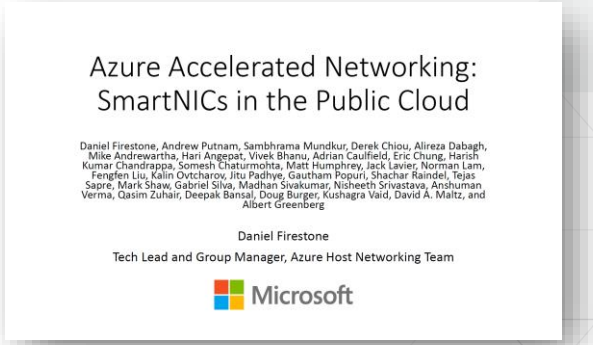
- Amazon Nitro, Microsoft Azure SmartNIC in every server (>2 million deployed)
- Use cases evolving, need for reconfigurable acceleration
- Cloud service providers want new efficiencies with the decline of Moore's Law
- Storage and compute acceleration in the NIC

© Copyright 2018 Xilinx

Virtualization and BareMetal trend favors FPGAs

Microsoft Azure SmartNIC





- FPGA enables support for >50Gbps networking
- Microsoft leadership for FPGA adoption
- Hyperscale deployment (>1M)

Aren't FPGAs much bigger than ASICs?	Not really, within 2x to 3x with targeted features
Aren't FPGAs very expensive?	Not really, compared to CPUs, DRAM and Flash
Aren't FPGAs hard to program?	Not really, hw/sw codesign provides hw performance with sw flexibility
Can FPGAs be deployed at hyperscale?	Yes, >1m servers with FPGA SmartNICs deployed
Isn't my code locked in to a single FPGA vendor?	No, Shell-role model allows portability

© Copyright 2018 Xilinx

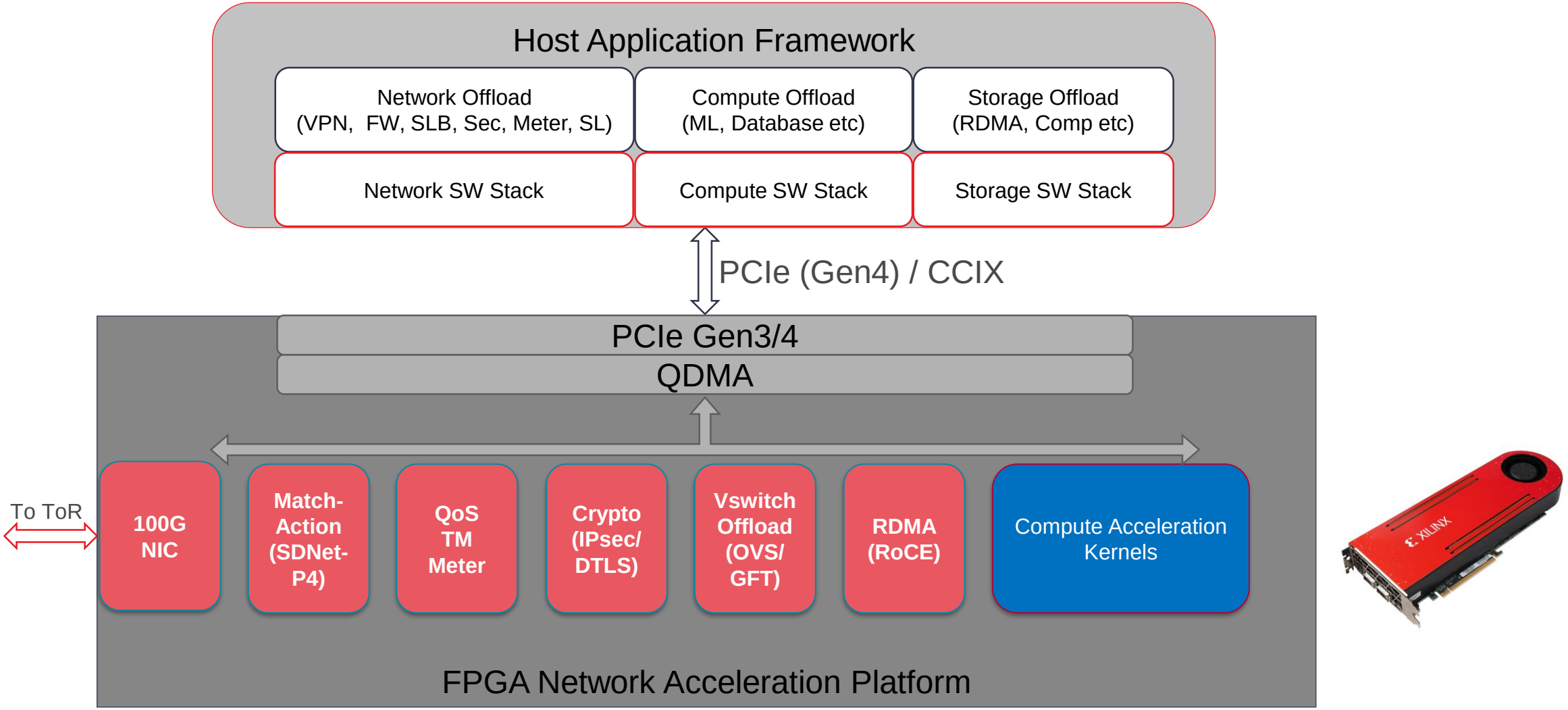
XILINX.

Agenda

- Networking Industry Trends
- Network Acceleration Platform & Usecases
- Key enabling IP, Tools & Technology

© Copyright 2018 Xilinx

Network Acceleration Platform



© Copyright 2018 Xilinx

XILINX

Typical Usecases

Basic Networking

- Basic NIC Offloads (LSO, TSO, RSS, Checksum)
- Tunneling Offloads (VXLAN, NVGRE)
- SR-IOV, MAC/VLAN Filtering
- QoS/Buffering

Network Acceleration

- P4/SDNet Programmable Data Plane
- OVS Acceleration
- Datapath Encryption/Decryption/ DPI
- In-Band Telemetry and Monitoring

Storage Acceleration

- NVMe, NVMe-oF, RoCEv2, NVMe-oTCP
- Storage Compression, Encryption, Dedup (RAID)
- Database Key Value Store and Analytics
- SQL Query Acceleration

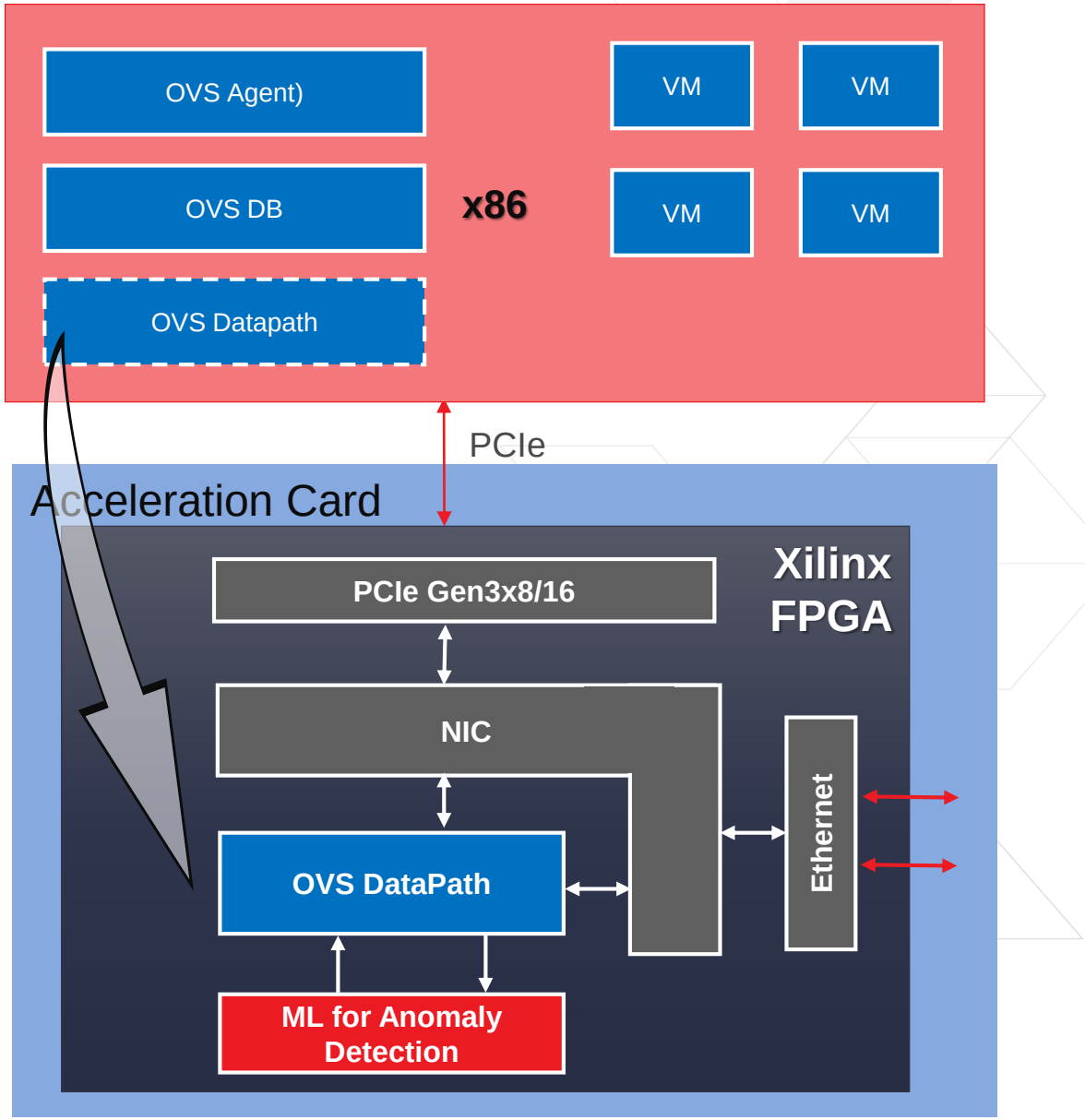
Compute Acceleration

- Machine Learning – Random Forest, K-means
- Video Encode, Decode, Scaling
- Deep Learning – CNN Inference
- HPC/Scientific

© Copyright 2018 Xilinx

Usecase - OVS Offload with ML

- **Problem: Malware detection on encrypted traffic being switched by OVS**
- **Solution:**
  - Redirect encrypted traffic to separate ML block for scanning
  - Feed header fields, packet length, inter-packet gap, packet rate, etc. into ML engine
  - Use trained decision trees to flag and tag or redirect suspicious traffic



openstack CLOUD SOFTWARE

OvS Open vSwitch

© Copyright 2018 Xilinx

XILINX

Agenda

- Networking Industry Trends
- Network Acceleration Platform & Usecases
- Key enabling IP, Tools & Technology
  - QDMA
  - SDAccel Streaming Platform
  - SDNet-P4
  - Typical SoftNIC

© Copyright 2018 Xilinx

PCIe QDMA IP

Converged Host Interface

- Stream, Memory Mapped, Network, Storage

Configurable

- Upto Gen3x16 support

2048 Queue sets

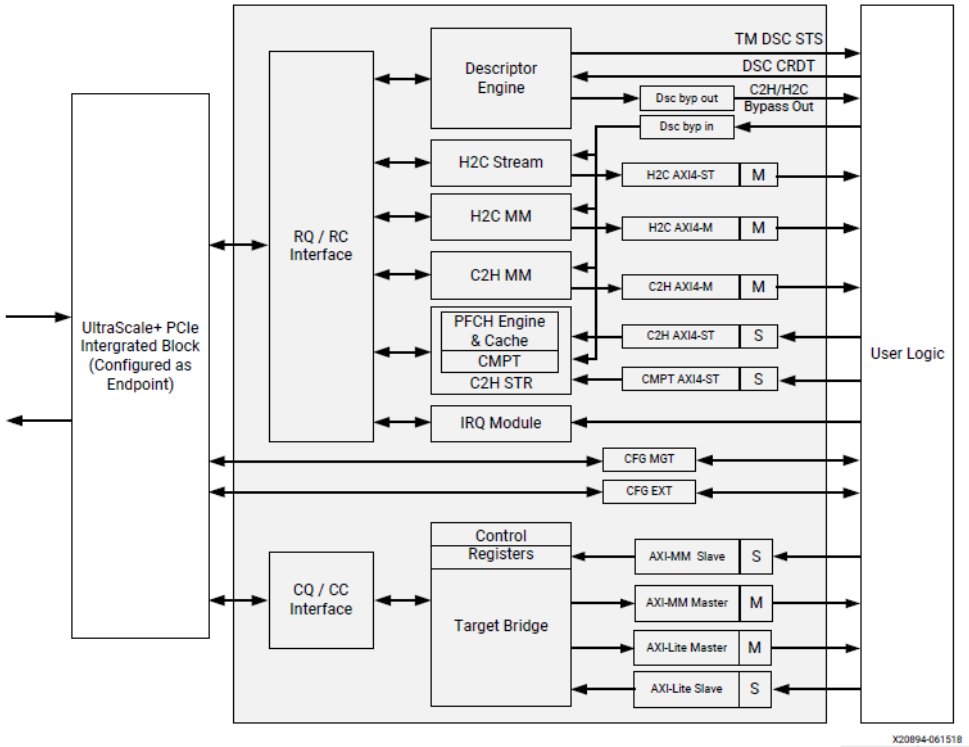
- H2C, C2H, CMPT

SR-IOV Support

- 4PF, 252VF

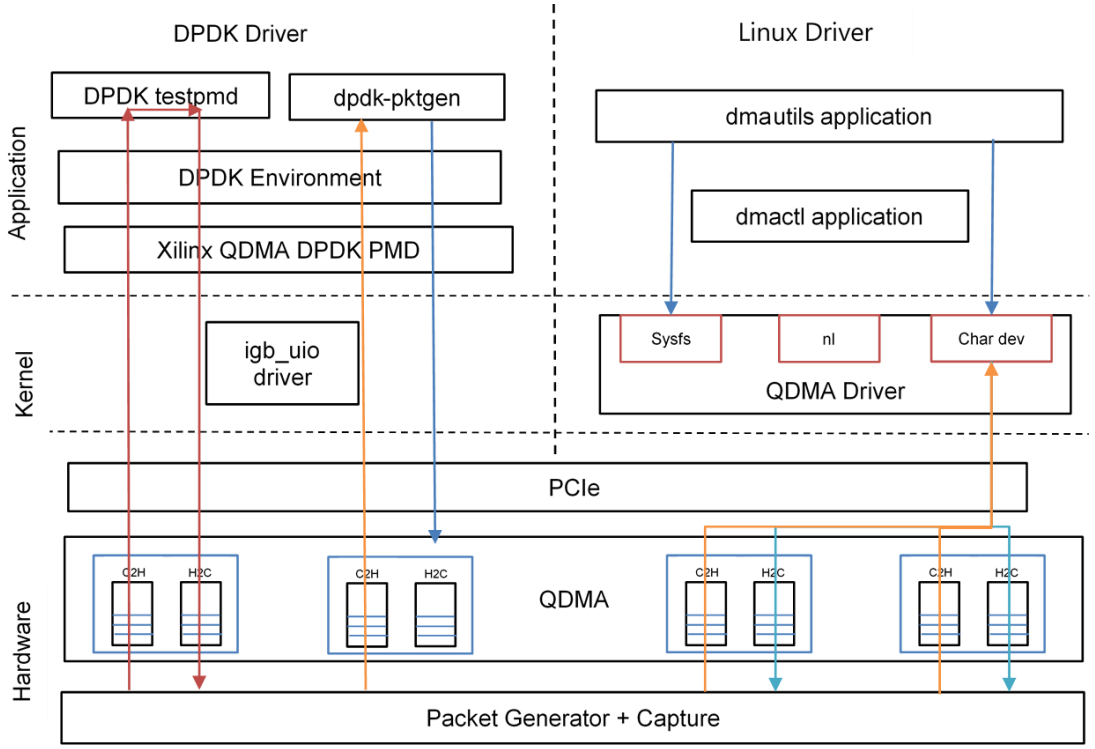
Customization with User Logic

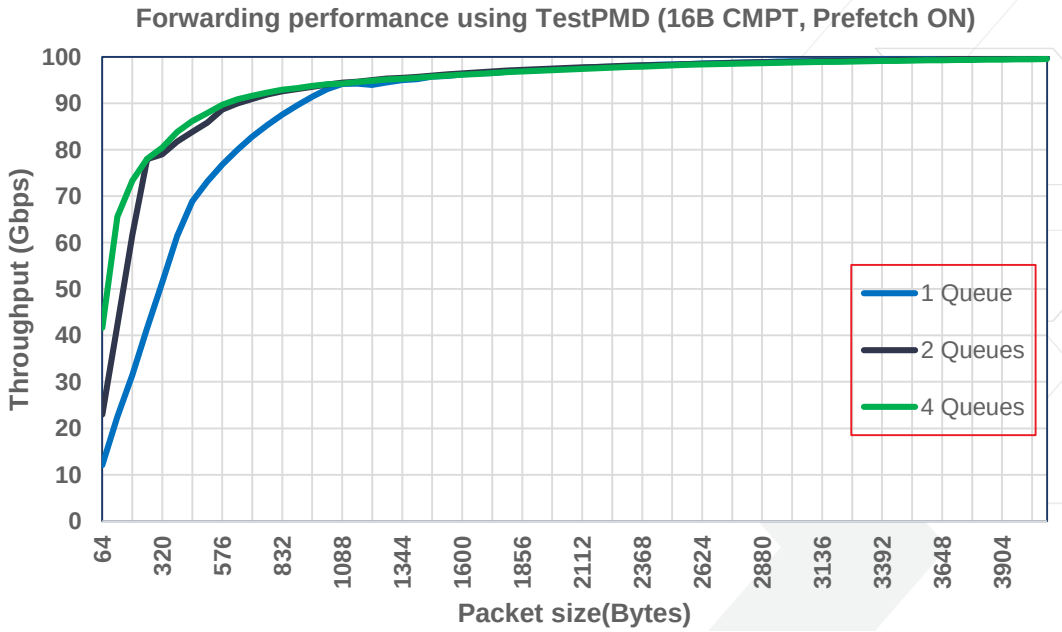
- Traffic Management, Virtual Switch, RDMA, NVMe-EP



© Copyright 2018 Xilinx

PCIe QDMA IP Performance





https://www.xilinx.com/products/intellectual-property/pcie-qdma.html#overview
https://www.xilinx.com/support/answers/34536.html

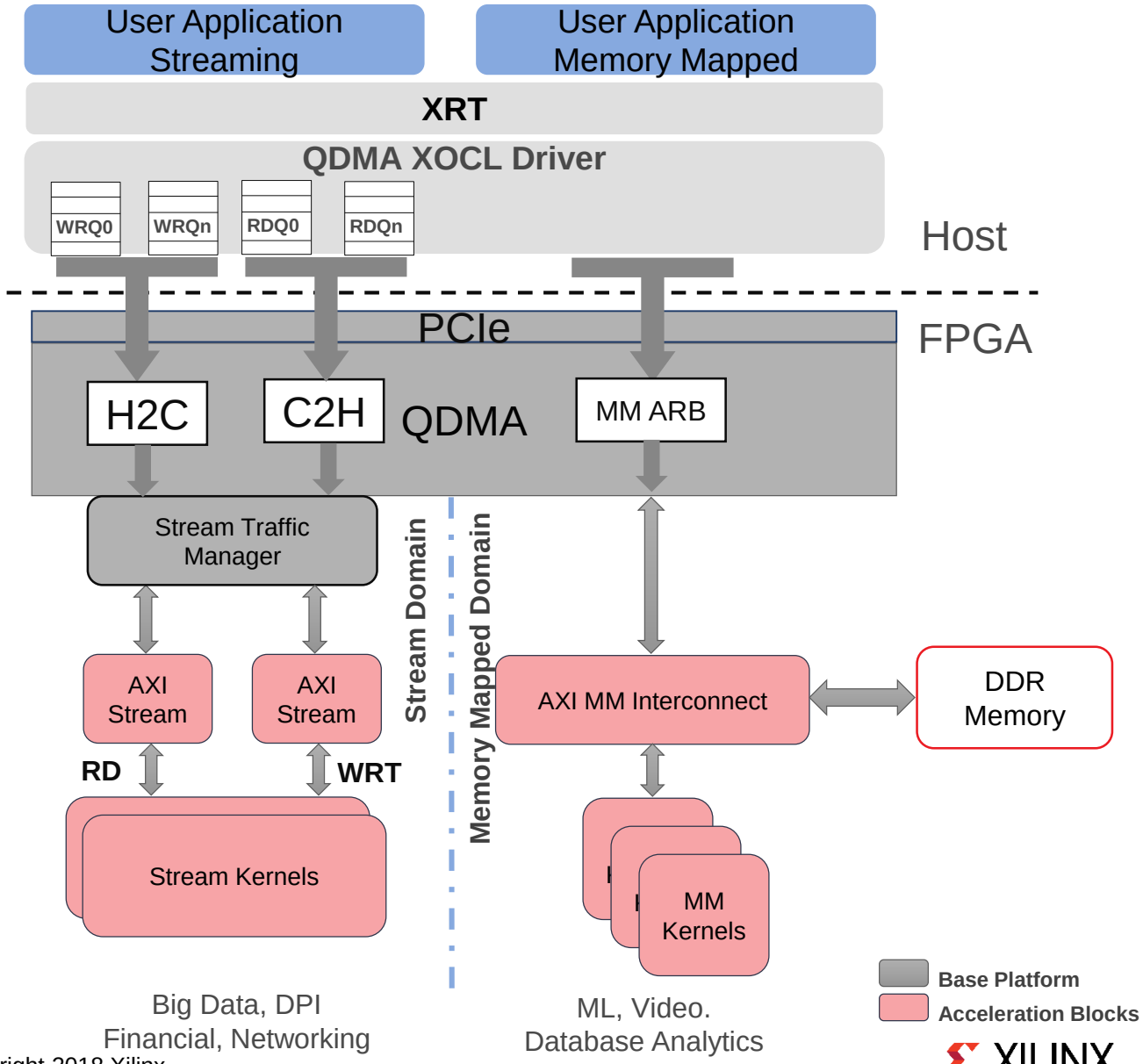
Leading edge Small Packet Performance exceeding 95Mpps

© Copyright 2018 Xil

* Forwarding performance >98Mpps for 64B packets with 8 Queues

SDAccel Streaming Platform

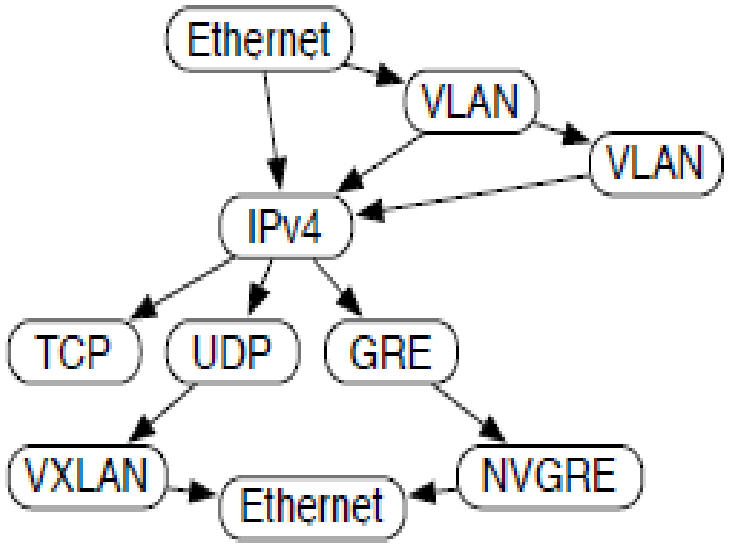
- Streaming Platform EA planned for 2018.3
- Kernels can Rd/Wrt streaming data from Host or to each other
- Stream Traffic Manager enables multiple streaming kernels



© Copyright 2018 Xilinx

Why SDNet?

Parser in P4: Data Center Usecase Example



(b) Data center.

```
parser MyParser(packet_in packet,
                out headers hdr,
                inout metadata meta,
                inout standard_metadata_t
smeta) {

    state start {
        transition parse_eth;
    }

    state parse_eth {
        packet.extract(hdr.eth);
        meta.eth_dmac = hdr.eth.dmac;
        transition select(hdr.eth.type) {
            VLAN_TYPE : parse_vlan;
            IPV4_TYPE : parse_ipv4;
            IPV6_TYPE : parse_ipv6;
            default   : accept;
        }
    }

    state parse_vlan {
        packet.extract(hdr.vlan.next);
        //meta.vlan_tpid = hdr.vlan.last.tpid;
        transition select(hdr.vlan.last.tpid) {
            VLAN_TYPE : parse_vlan;
            IPV4_TYPE : parse_ipv4;
            IPV6_TYPE : parse_ipv6;
            default   : accept;
        }
    }
```

```
state parse_ipv4 {
        packet.extract(hdr.ipv4);
        packet.extract(hdr.ipv4opt, (((bit<32>)hdr.ipv4.hdr_len - 5) * 32));
        verify(hdr.ipv4.version != 4, error.IpVersionNotSupported);
        meta.ipv4_src = hdr.ipv4.src;
        transition select(hdr.ipv4.protocol) {
            TCP_PROT  : parse_tcp;
            UDP_PROT  : parse_udp;
            GRE_PROT  : parse_gre;
            default   : accept;
        }
    }

state parse_tcp {
        packet.extract(hdr.tcp);
        packet.extract(hdr.tcpopt, (((bit<32>)hdr.tcp.dataOffset - 5) * 32));
        meta.tcp_seqnum = hdr.tcp.seqNum;
        transition accept;
    }

state parse_gre {
        packet.extract(hdr.gre);
        packet.extract(hdr.greopt, (((bit<32>)hdr.gre.c_flag * 32) +
                                    ((bit<32>)hdr.gre.r_flag * 32) +
                                    ((bit<32>)hdr.gre.k_flag * 32) +
                                    ((bit<32>)hdr.gre.s_flag * 32)));
        meta.gre_proto = hdr.gre.proto_type;
        transition select(hdr.gre.proto_type) {
            NVGRE_TYPE : parse_nvgre;
            default    : accept;
        }
    }
```

P4 enables networking design to be completely programmable, top-to-bottom, soup-to-nuts*

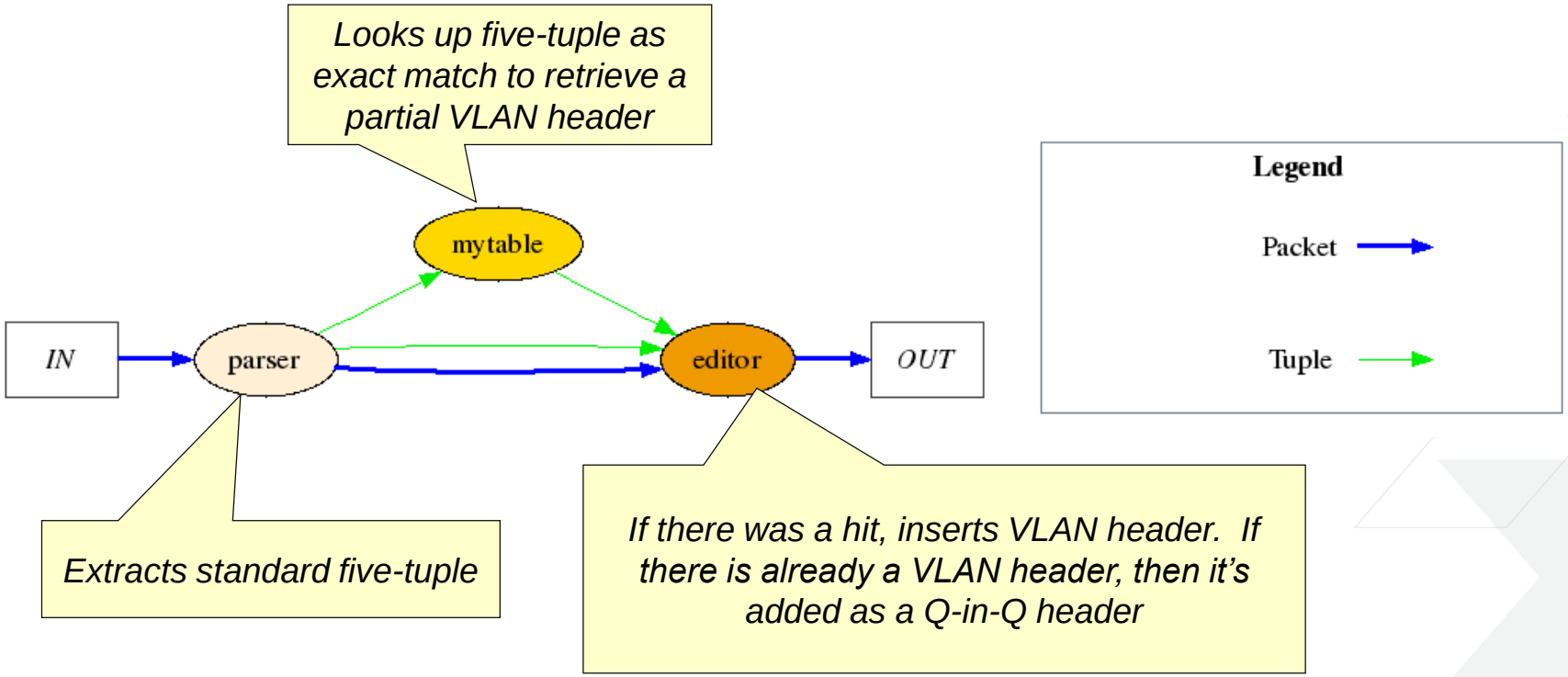
© Copyright 2018 Xilinx

* P4.org

XILINX.

100Gbps Packet Processing with SDNet

Resource Utilization for standard five tuple example

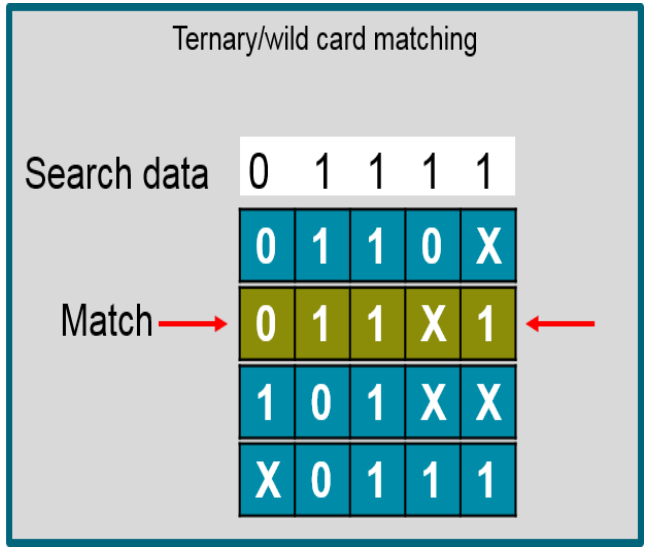


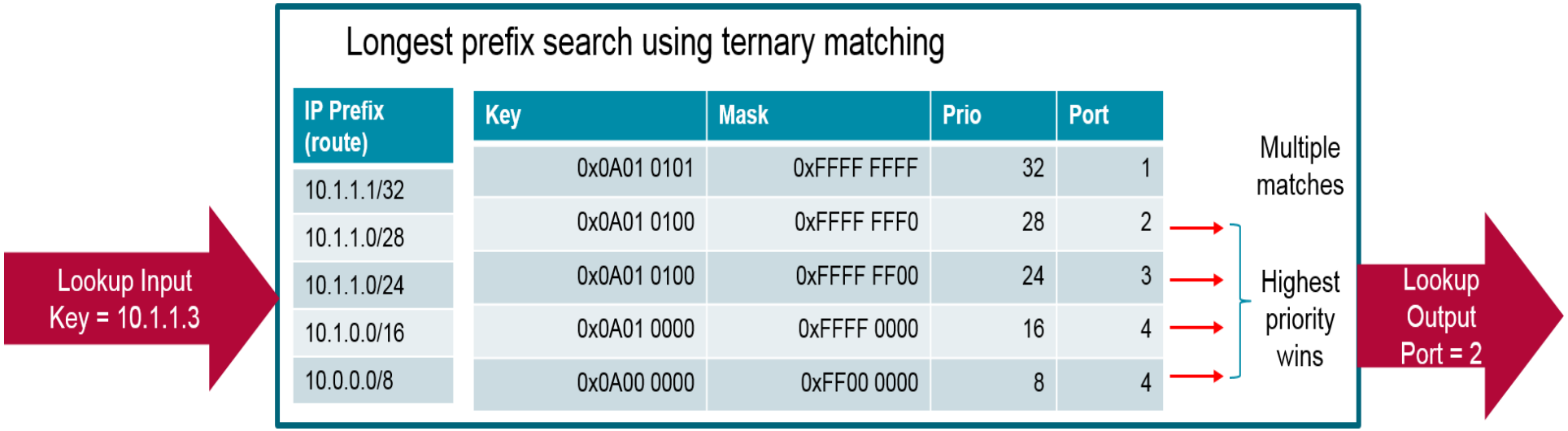
Tool	Language	Table size	k LUTs	k FFs	BRAM	URAM	Rate
SDNet 2018.1	SDNet – PX	64k EM (64Kx104x16)	19.1	23.7	27	56	100 Gb/sec
SDNet-P4 2018.3	SDNet - P4 (estimate)	64k EM (64Kx104x16)	8K	10K	27	56	100 Gb/sec
SDNet-P4 2018.3	SDNet - P4 (estimate)	Parser only	1.1K	2.5K	-	-	100 Gb/sec

© Copyright 2018 Xilinx

Xilinx Algorithmic Lookup Features

- Exact matching for MAC addresses and flows
- Multi-field wild card matching for ACLs
- Longest prefix matching for IP routes
- Priority decoding for multiple matches
- Ternary and semi Ternary masking
- TCAM capacity of each BRAM = 20Kb, URAM =160Kb





© Copyright 2018 Xilinx

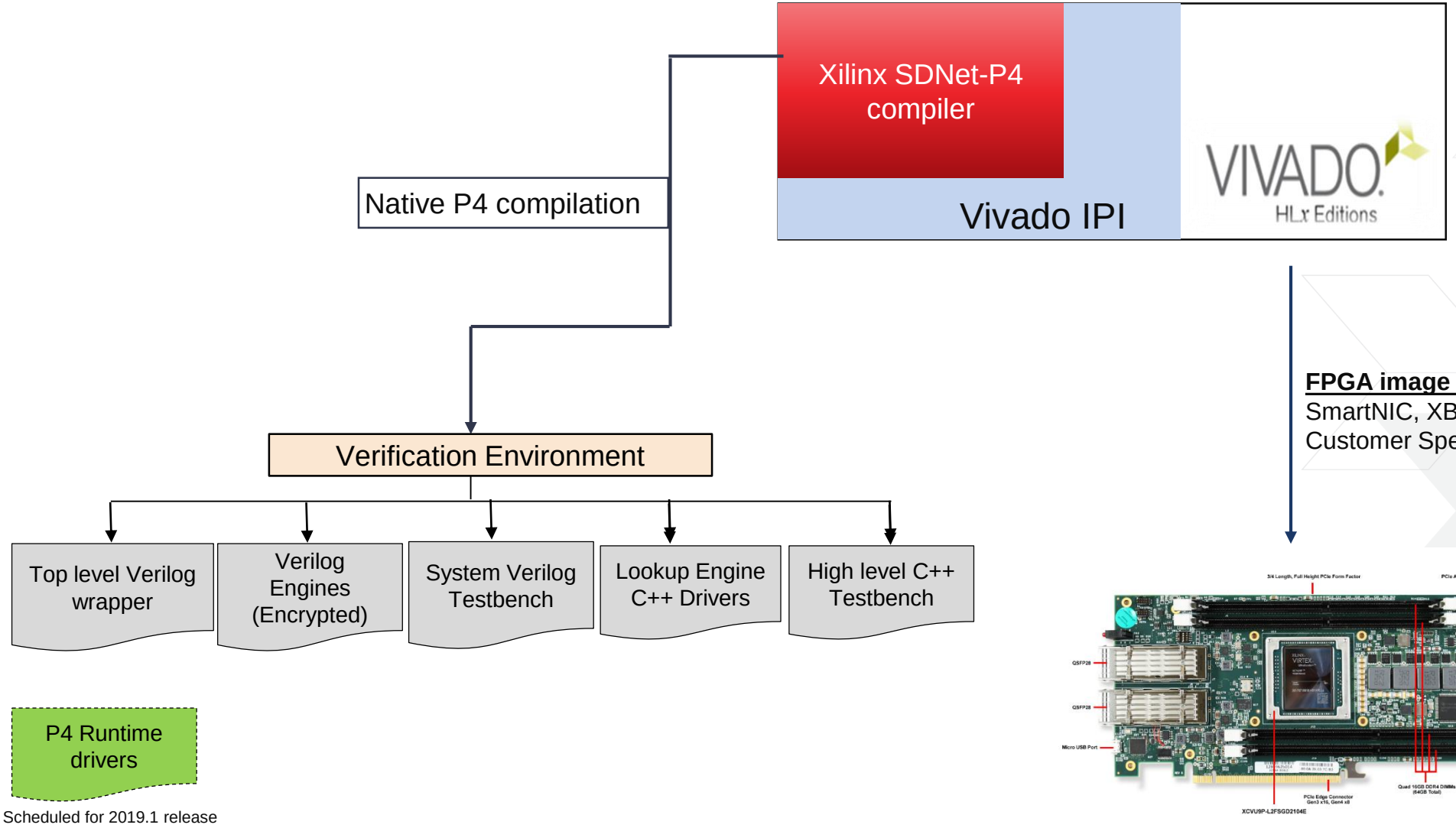
SDNet Algorithmic Lookup Solution

BRAM/URAM	DDR	HBM
600 MSPS	External DRAM based	Up to 8GB on chip DRAM for lookups
20M/Sec inline and 100K/Sec SW updates	Millions of ACL/LPM/EM entries	Millions of ACL/LPM/EM
40-60 clock cycle latency (600 MHz)	Integrated with Xilinx DDR controller IPs	HBM as result memory
EM/LPM/ACL : SDNet 2018.3	EM: SDNet 2019.1 LPM/ACL: SDNet 2019.2	EM/LPM/ACL : SDNet 2019.2

© Copyright 2018 Xilinx

Xilinx SDNet-P4 v2018.3

Scheduled for release in November 2018



© Copyright 2018 Xilinx

SDNet Roadmap

SDNet 2017.3	SDNet 2018.1	SDNet 2018.3	SDNet 2019.1	SDNet 2019.2

• **PX** Support
• **System Builder**
• VU9P SLRD (VCU118)
	• Minor release
	• **200Gbps** throughput with five tuple design on Ultrascale+ devices
		• **P4** as primary development language
		• **New URAM-based Search IPs** for
		  • EM,
		  • LPM
		  • TCAM
		• **2x-3x** better Resource and Latency Improvements
			• **DRAM** support for EM Search IP
			• New P4 runtime APIs
				• Search IPs with **DRAM** (HBM and DDR) for **ACLs and LPM**
				• EA Support for **7nm** devices

© Copyright 2018 Xilinx

Typical Soft NIC Feature Summary

- SDNet-P4 enables NIC datapath
- Ethernet
  - 100GbE/50GbE/25GbE
- PCIe
  - Gen3x16, Gen3x8
- OS support
  - Windows
  - Linux & DPDK
  - VMWare

- Stateless NIC offloads
- Packet flow steering/Filtering
- Tunneling offloads
- Server Virtualization
- Server Manageability
- Remote Boot

© Copyright 2018 Xilinx

Summary

© Copyright 2018 Xilinx

Summary

- FPGAs enable Adaptable Network Acceleration
- Hardware Performance with Software-level programmability
- Customized solutions for 100Gbps and beyond
- Rich set of IP, Tools and Boards

© Copyright 2018 Xilinx

Adaptable.
Intelligent.

XILINX

© Copyright 2018 Xilinx