SmartNICs SUMMIT
April 26-28, 2022
DoubleTree by Hilton San Jose
SmartNICsSummit.com

Implementing a Fourth Generation SmartNIC

Steve Zagorianakos & Johann Tonsing

Corigine & NETRONOME

Key features of fourth generation SmartNICs

Programmability

- Must have the flexibility and ease-of-use to address current and future networking workloads
- Flexible C programmable data path
- Offload from server CPUs is required to support modern distributed applications

Scalable processing

- Standard Linux processor
- Many Data Path processors
- Custom processors: to support hardware acceleration engines

Scalable high-performance network and PCIe interfaces

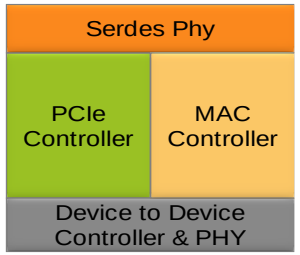
- Multiple network ports
- Multi lane 100G serdes support
- Multi PCIe controller: root complex and endpoint support

Large DDR memory ports on SmartNIC

Scalable architecture, implemented as *Chiplets*

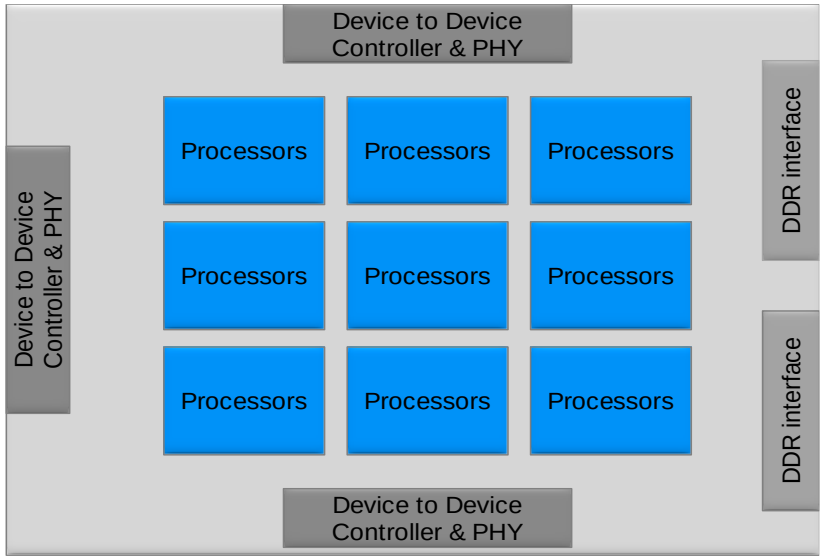
IO Chiplet:

- Multiple 100G serdes, selectable between network and PCIe
- High speed network up to 400G interfaces
- PCIe gen5 interface: 4 controllers, up to 16 lanes total - both root complex and endpoint
- High performance Device to Device interface



Processing Chiplet:

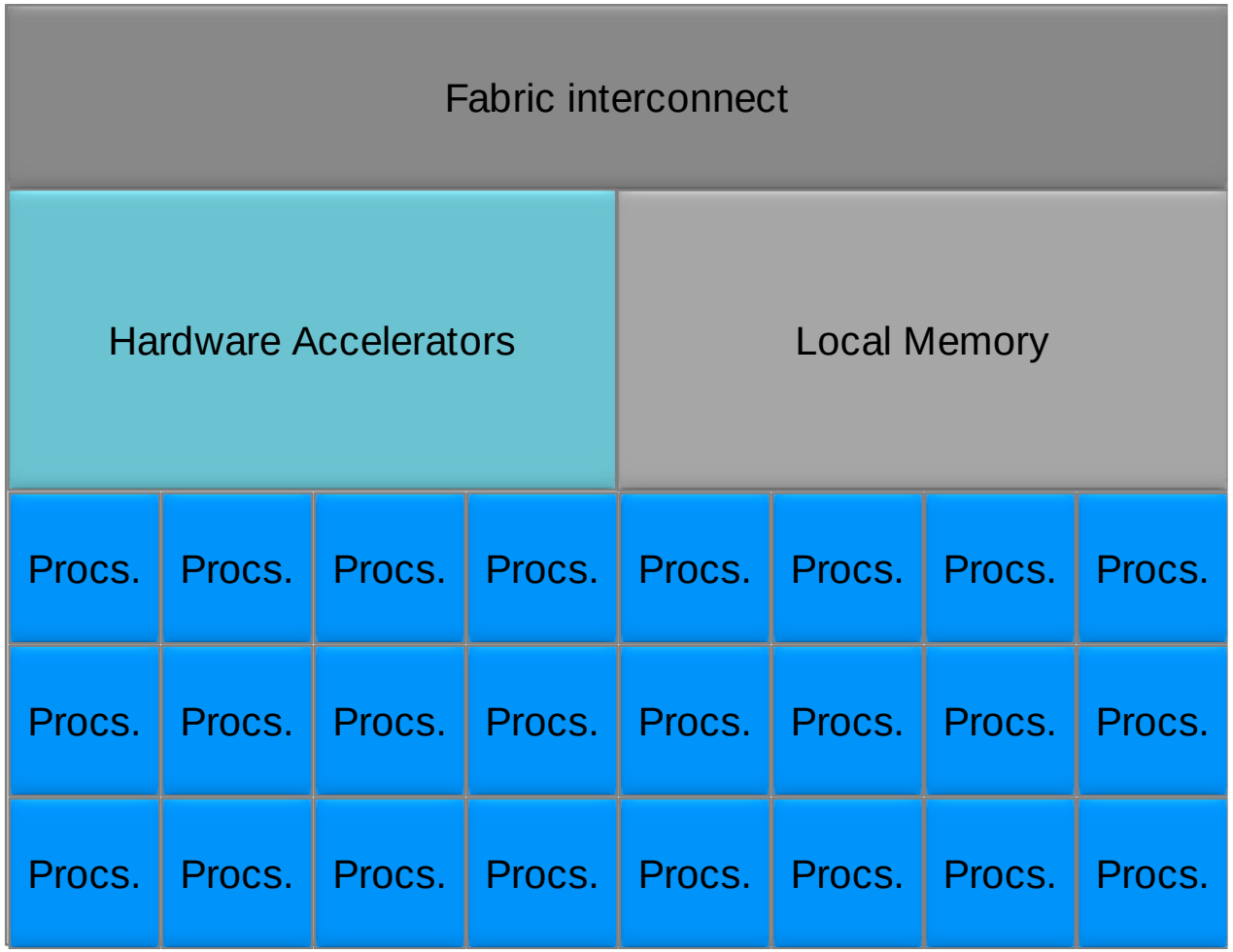
- Full custom: multi-ALU used for operations like characterize, load balance, initial packet match and transactional memory
- Standard RISC-V small cores with custom modifications to support network processing, access to a small ALU and hardware acceleration engines
- Standard RISC-V large multi-core running Linux, for slow path and control plane
- Two DDR channels with last level cache and transactional engines: atomic, CRC, queueing, stats, CAM, lookup…
- High performance intra-chiplet fabric
- Three high performance Device to Device interfaces to connect *chiplets*



RISC-V Processor Block

- Enhanced for processing network packets
- Hardware Acceleration: Crypto, Atomics, CRC, Queueing, Packet delivery to assigned processor
- Processors made up of a small RISC-V core, ~ 128 cores per block
- High performance block interconnect fabric

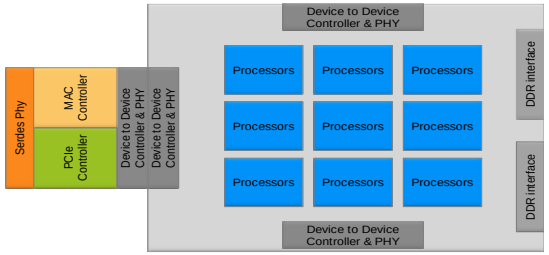
Scales to thousands of cores per chip!



Expandable product levels: utilizing multiple *Chiplets*

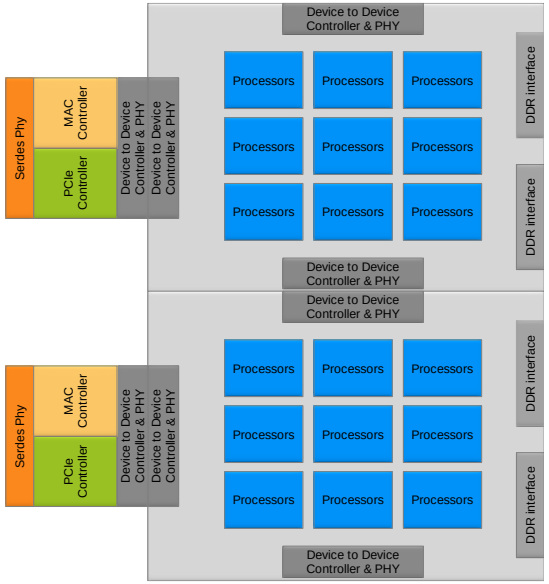
Entry-level performance

- 1x I/O, 1x Processor *Chiplets*
- 16 lanes, 4 PCIe controllers
- 4 lanes support 4x25/100G, 2x200G, 1x400G networking
- Zero to two DDR interfaces



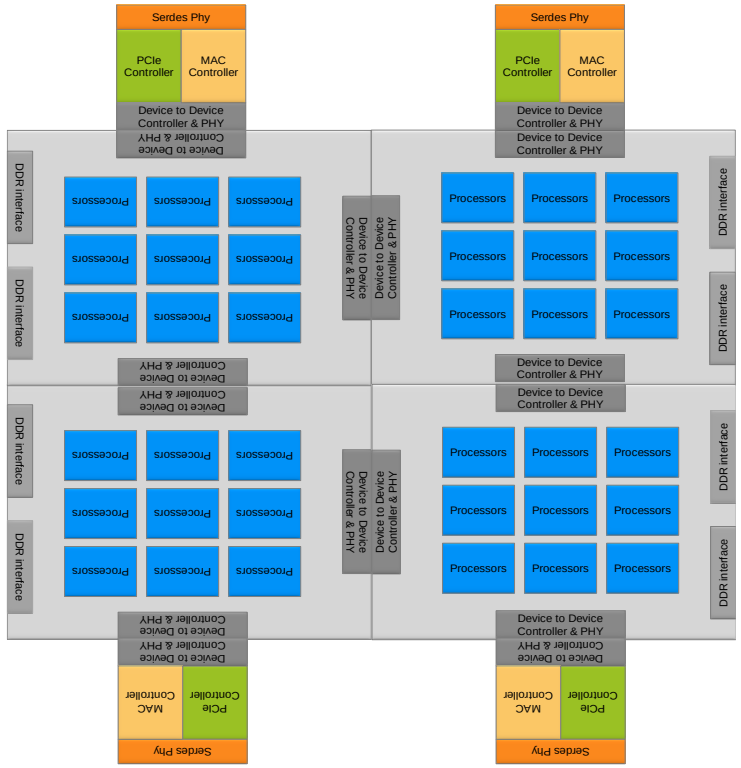
Mid-range performance

- 2x I/O, 2x Processor *Chiplets*
- 32 lanes, 8 PCIe controllers
- 8 lanes support 8x25/100G, 4x200G, 2X400G networking
- Up to four DDR interfaces



High-end performance

- Up to 4x I/O, 4x Processor *Chiplets*
- 64 lanes max for PCIe (16 controller), w/ 16 max for network
- 48 lanes for PCIe w/ 32 lanes for Network
- Up to eight DDR interfaces



Comprehensive and Proven SmartNIC/SmartSwitch Platform

Application Examples	Virtual Switching and Routing	VXLAN, GRE, GENEVE, GTP, MPLS, SRv6	ACLs, Sec. Groups, DDoS Prot.	SSL, kTLS, IPsec, NGFW, IPS, DLP, DPI	Storage Virt. Offload	Traffic Steering + Load Bal.	vProbes, In-band Telemetry	QoS / CoS, Congestion & Tail Latency Reduction

Basic NIC Features

Offload NIC Features

Programmable NIC Features

Data Plane Programming Tools for Custom Feature Additions

P4, C, eBPF/XDP-based Programming; Island + Chiplet-Based Architecture

Virtualization, Security, Visibility, Storage - SDN, SDDC, SD-WAN

Open vSwitch / Contrail vRouter, IPsec / TLS, Storage Virtualization, QoS / TM

Basic Firmware + Device Drivers (Linux, VMware, Windows…)

Stateless Offloads, SR-IOV, DPDK, VirtIO, vDPA, RDMA

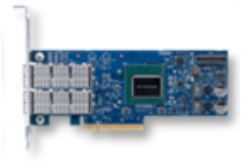
Silicon + NIC/Switch Family (PCIe, OCP…)

1-4 dataplane processor and 1-4 I/O chiplets

Ethernet: 25G to 400G - up to 32 ports

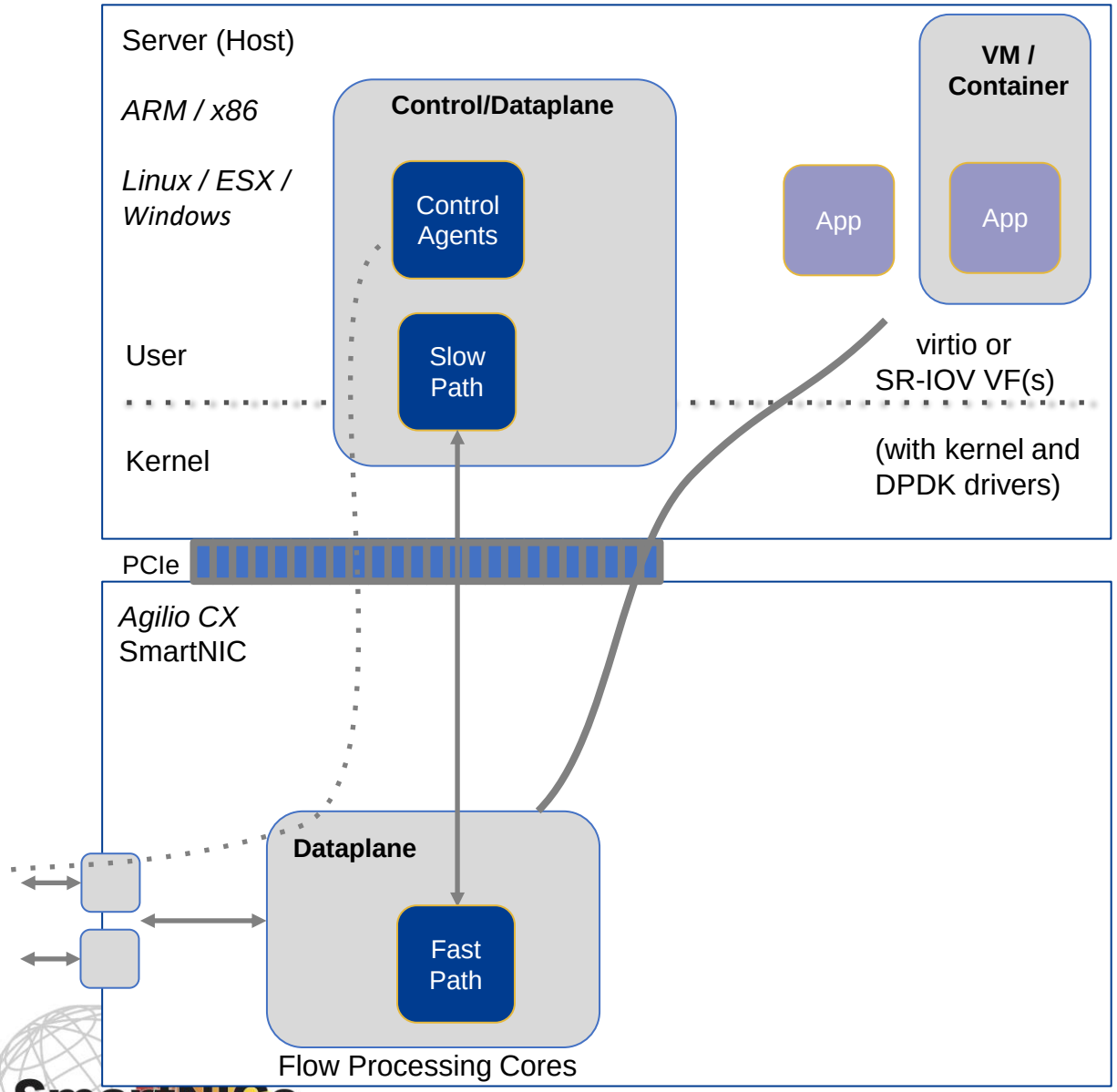
PCIe: gen3/4/5 - up to 64 lanes

Multicore control plane + slowpath processor running Linux => bare metal NIC



SmartNICs SUMMIT

SmartNIC with Offloaded Switch and SR-IOV / virtio



Characteristics:

- High flexibility and good security
- Supports virtio drivers for VM migration
- High performance (in SR-IOV and virtio modes)
- Minimal host CPU overhead with standard control plane

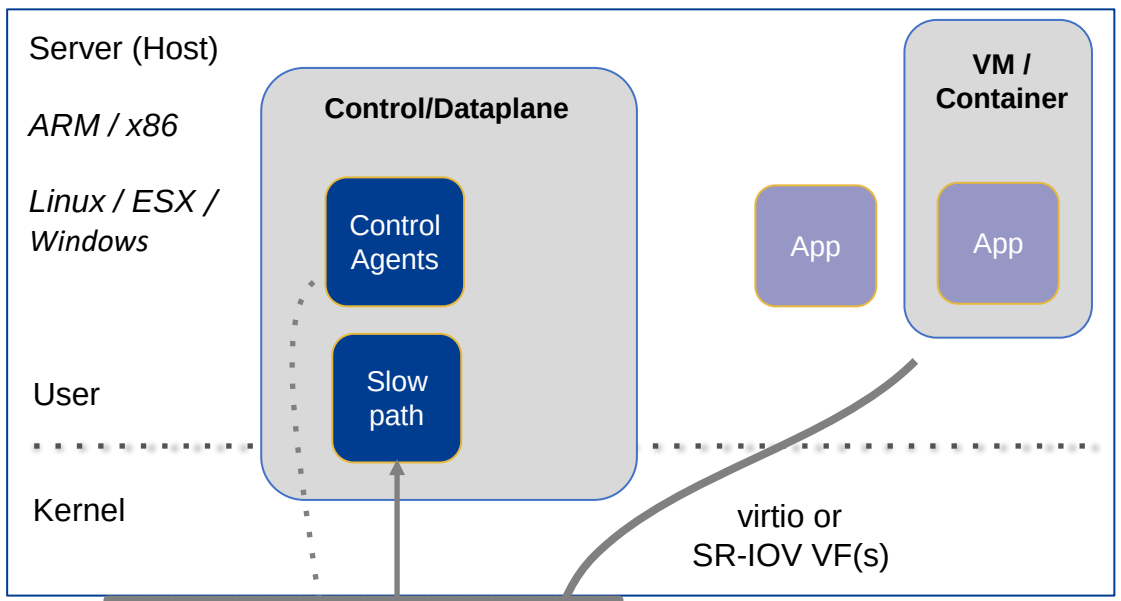
Smart Dataplanes: *Upstreamed (RHEL + RHOSP, Ubuntu…)*

- Open vSwitch (OVS) — directly or via OVN
  - Kernel (TC) and DPDK (RTE_FLOW) variants
  - Optional: connection tracking (firewall) with NAT
  - Optional: rule-based metering
- Contrail vRouter + Tungsten Fabric
  - Includes firewall capability
- XDP - eBPF
- IPsec, Wireguard, SSL/TLS - AES/SHA, SM3/SM4
- Storage virtualization offload (RoCEv2, NVMe over TCP…)
- *Custom - implemented in C and/or P4*

Traditional Dataplane:

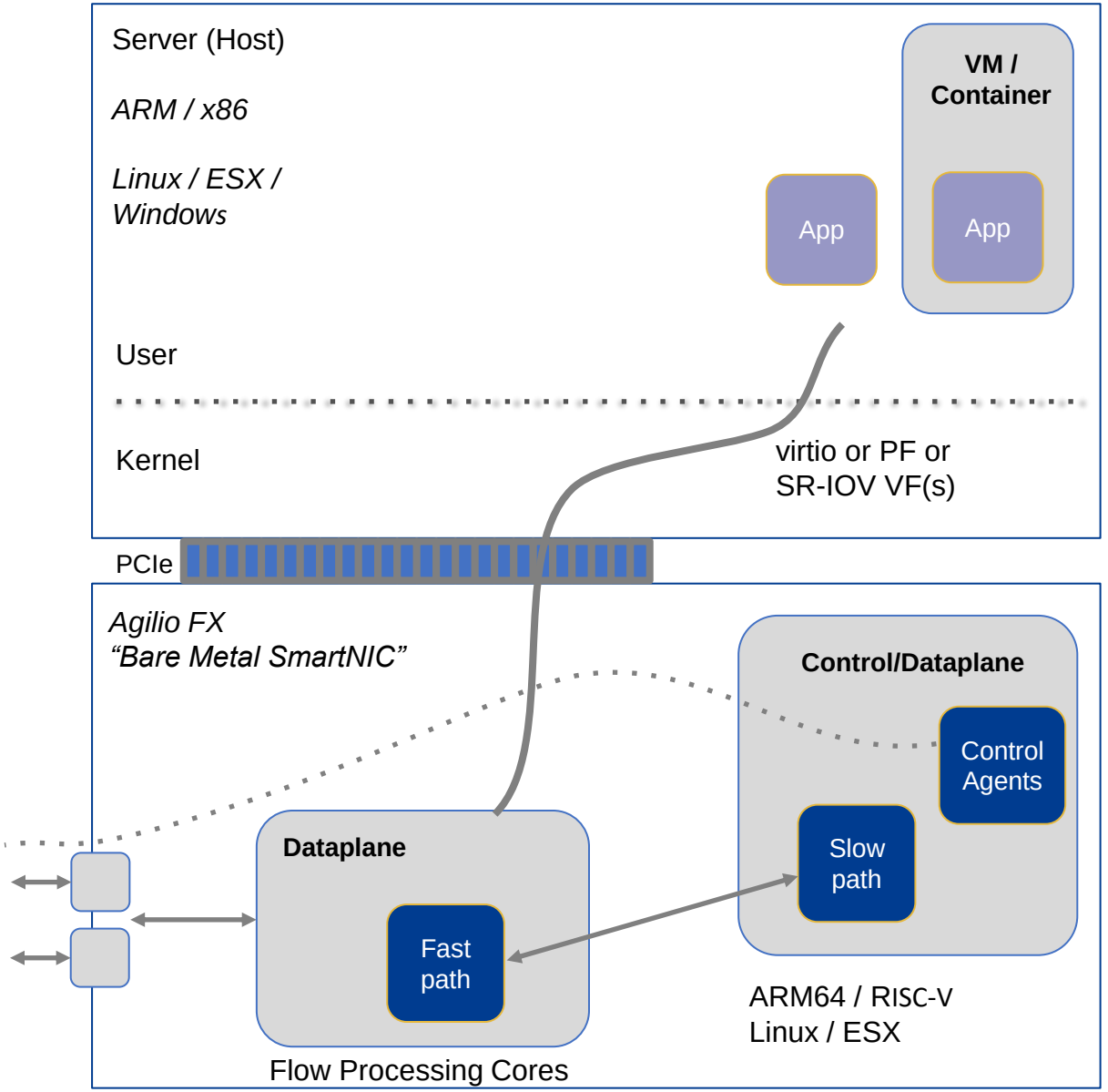
Standard basic NIC firmware + drivers permit standardizing on one hardware platform - enabling smart capabilities as needed

Server Offload SmartNIC vs. Bare Metal SmartNIC



FX series:

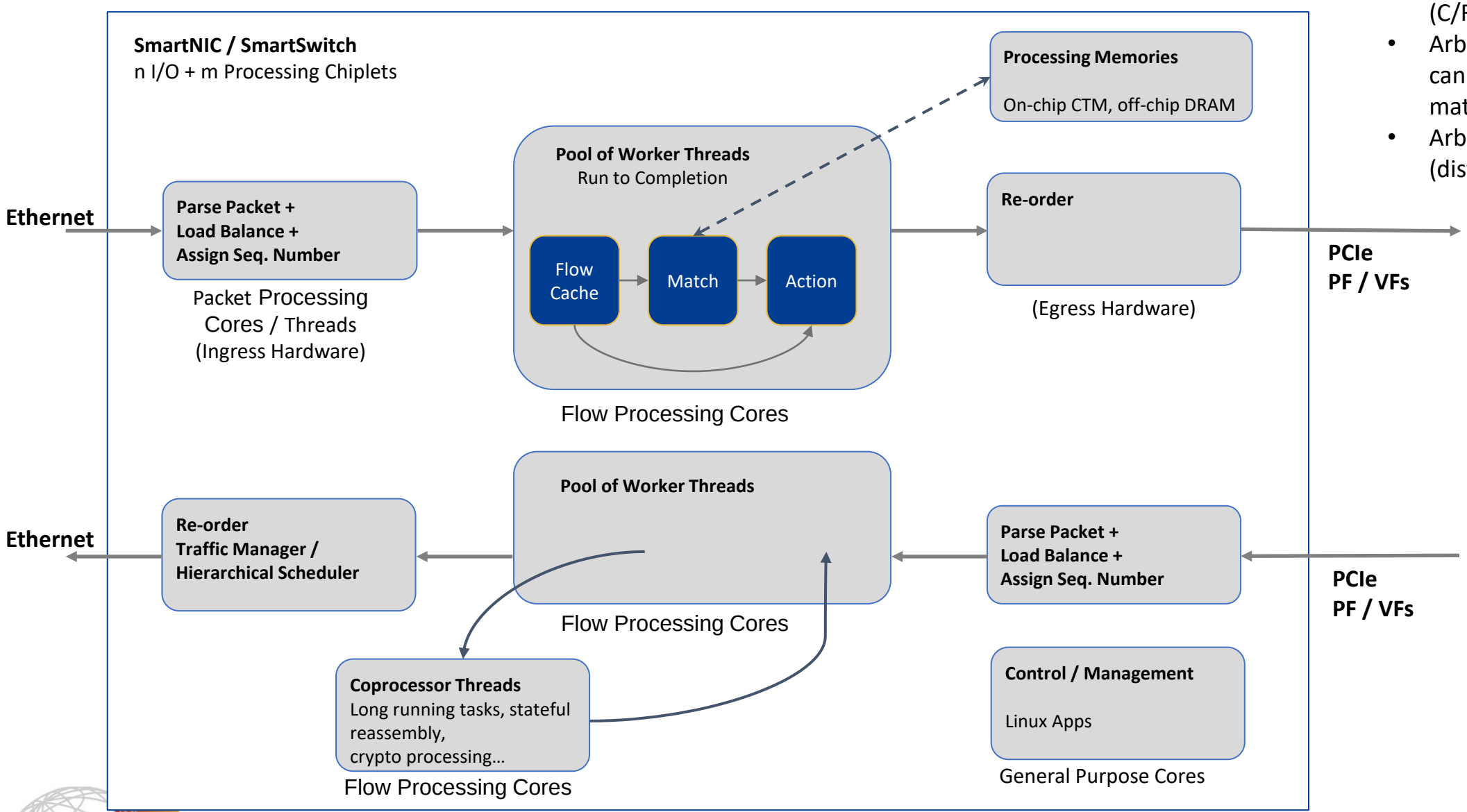
- High performance
- Low Power
- Easy to program
- Data plane and Control plane isolated from host



Programming Model

Characteristics:

- Standard compilers / languages (C/Rust etc. - GCC/LLVM)
- Arbitrary user mode code can run - not limited to e.g. match/action (P4 or OVS)
- Arbitrary topology of software (distributed over chiplets)



Thank You

- Questions?