Fast Packet Processing Methods

James Won-Ki Hong, Seyeon Jeong, Jian Li

Dept. of Computer Science & Engineering

POSTECH

http://dpnm.postech.ac.kr/~jwkhong

jwkhong@postech.ac.kr

Outline

- Fast Packet Processing Methods
- Datapath Composition
- OpenStack Introduction

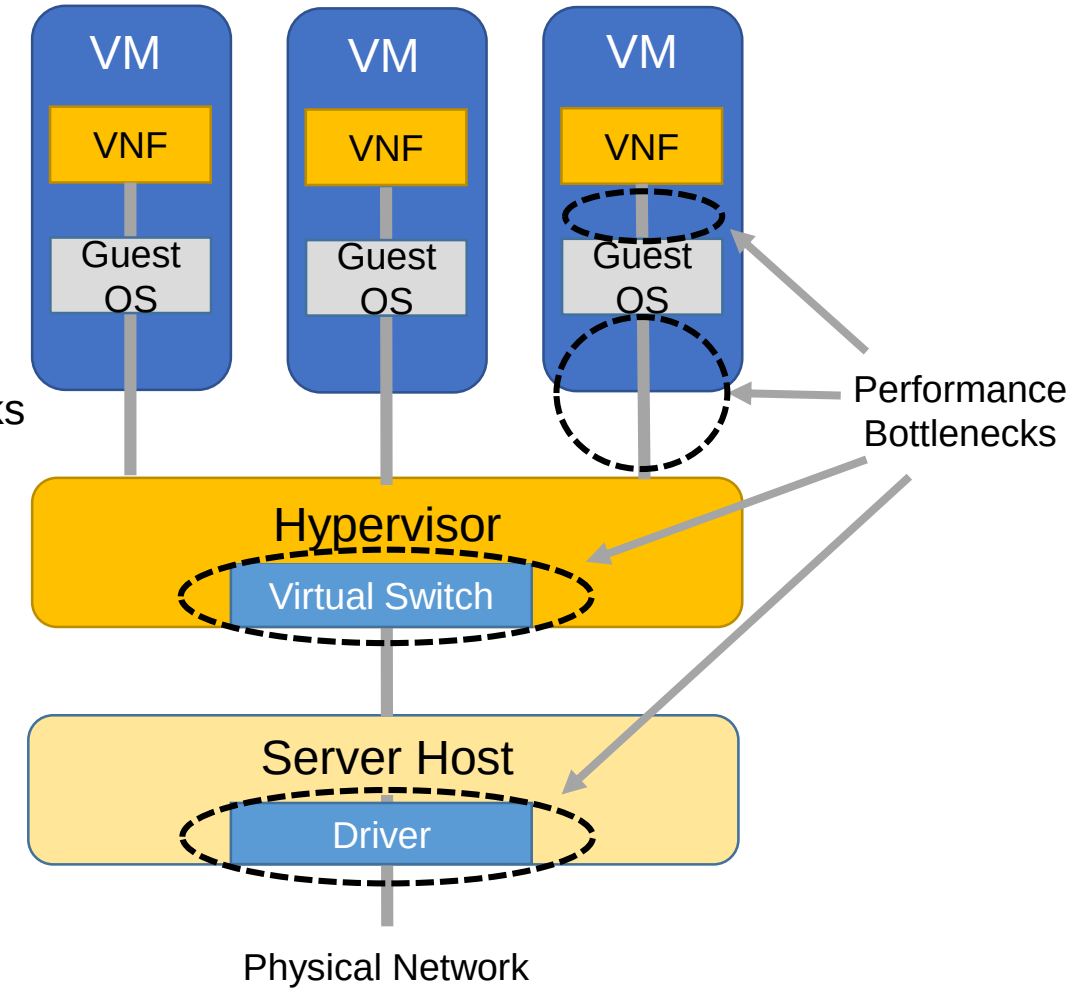
Software Packet Switching

❖ NFV

- Virtualizes network functions and decouples them from specialized H/W
  - Reduce costs and time to market
  - Allows a network function to run as a VM
    on commodity equpment (server virtualization)

- Server virtualization needs software-based data plane for packet processing (Virtual switch)
  - This concept involves several potential performance bottlenecks
  - Large amounts of CPU consumed by packet processing leave much less CPU for executing the network function

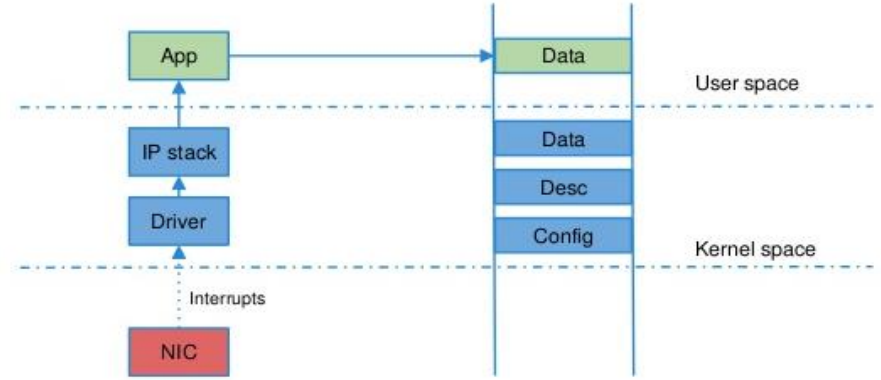
- We need to improve the performance through fast packet processing methods in S/W or H/W

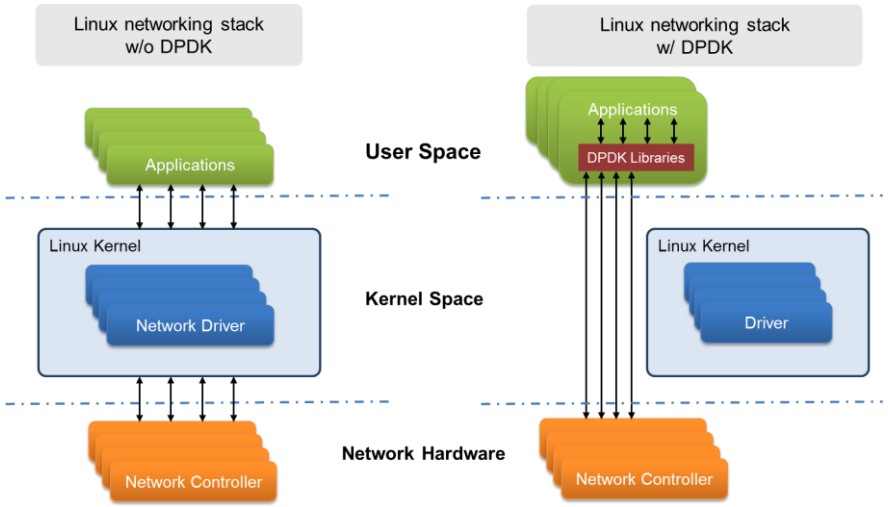


❖ DPDK (Data Plane Development Kit)

- Existing networking stack in Linux
  - Separates packet processing into kernel space (TCP/IP) and user space (App.)
  - Involves per-packet CPU interrupts (frequent context switching)
  - High resource consumption due to data copy b/w kernel and user space

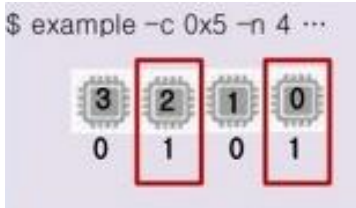
- A set of software libraries for high speed packet processing
  - Packet processing in user space, not kernel networking stack
  - Support for CPU affinity, huge page, multi-queue, etc.



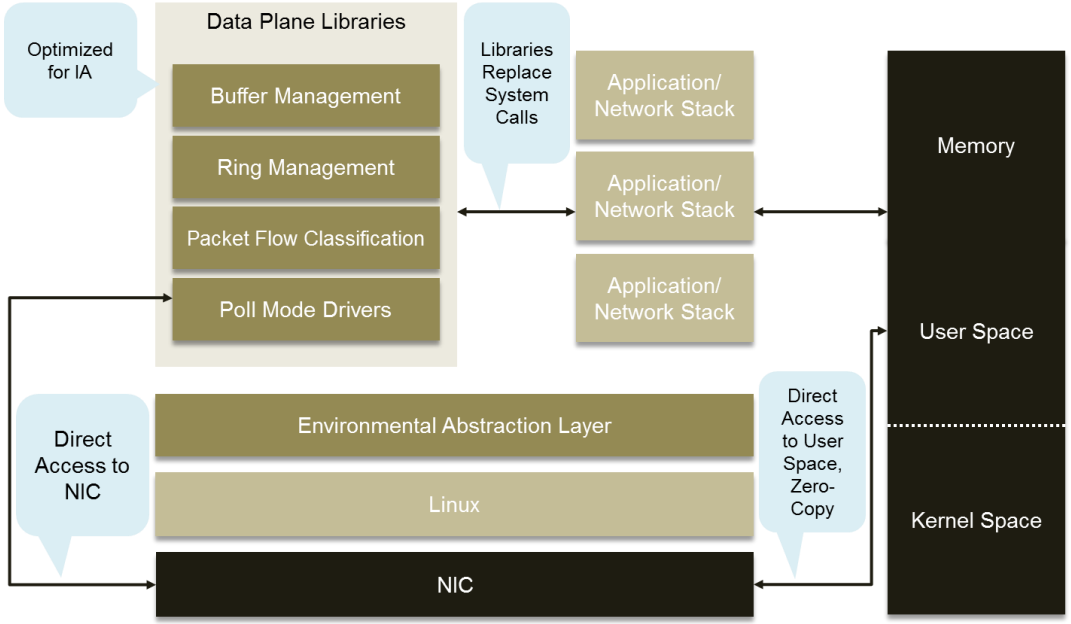


❖ DPDK Features

- Environment Abstraction Layer (EAL)
- Poll Mode Drivers (PMD)
- Memory Manager
- Buffer Manager
- Queue Manager
- Packet Flow Classification



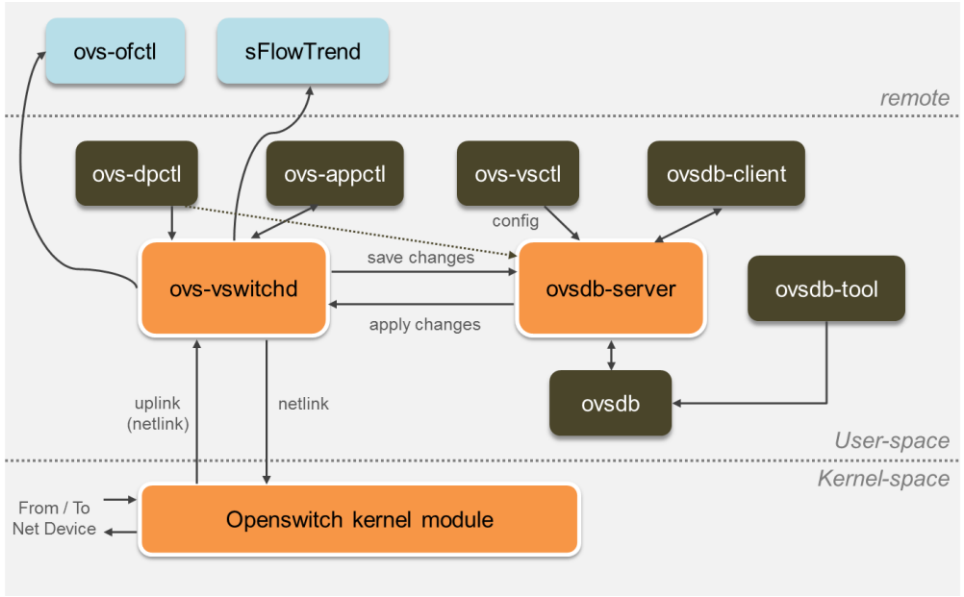
-c CPU_core_mask (hex)
-n #_of_memory_ channels



Open vSwitch

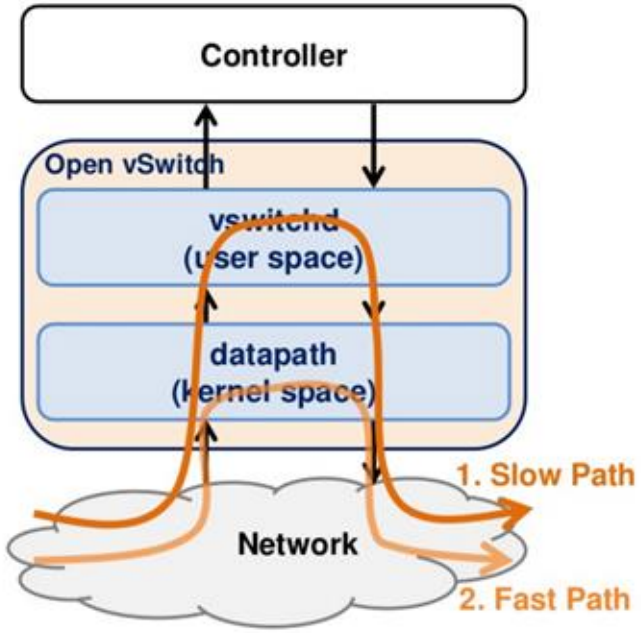
❖ Open vSwitch (OVS)

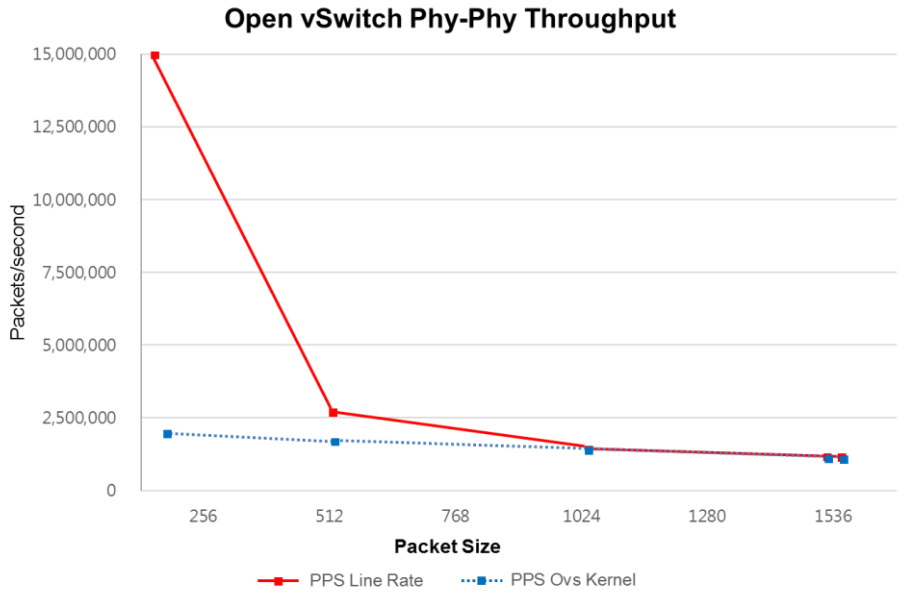
- A multi-layer virtual (software) switch under Apache 2.0
- Supports various networking services and SDN control mechanism through management protocols
  - Support for IPv6, tunneling, NIC bonding, etc.
  - OpenFlow, OVSDB, NetFlow, sFlow, (R)SPAN, etc.
- Key components
  - kernel module : forwarding, de/encapsulation, cache lookup
  - ovs-vswitchd : overall forwarding logic management (from remote)
  - ovsdb-server : an access to configuration details (ovsdb)



❖ Open vSwitch

- Packet forwarding
  - If any matching rule exists (cached), packets go through kernel module → Fast path
  - Otherwise, delay packets and look up matching rules in ovs-vswitchd → Slow path
- Performance degradation in slow path
  - Throughput, PPS, delay
  - Due to memory copy b/w kernel and user, lookup event handling, ...





Source: http://openvswitch.org/support/ovscon2014/17/1630-accelerating-with-dpdk.pptx

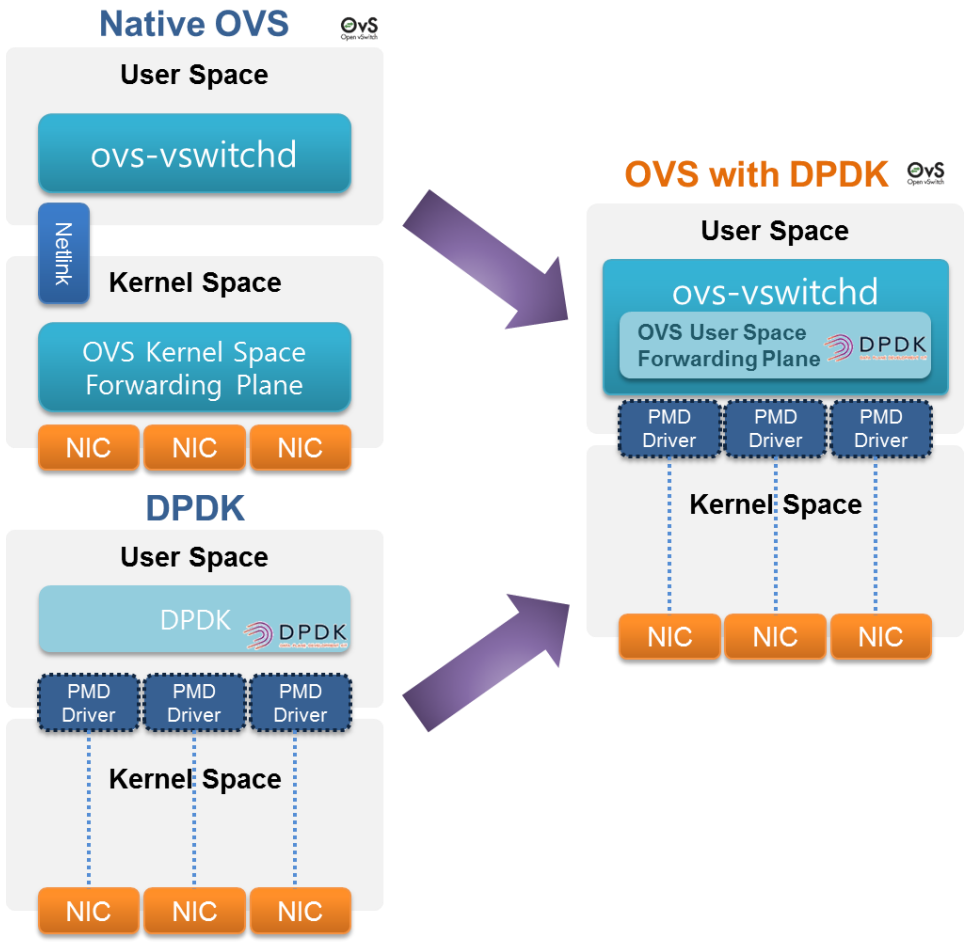
OVS-DPDK

❖ Open vSwitch with DPDK (OVS-DPDK)

- Started as Intel DPDK vSwitch
- Upstreamed to OVS and stably supported since OVS 2.4

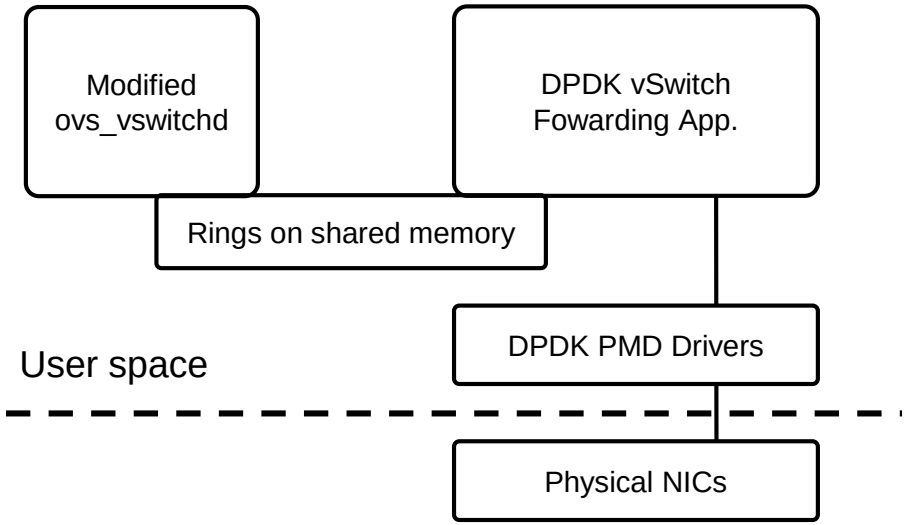
❖ Features

- DPDK-based implementation
  - Running based on EAL setups
- vHost-user port (Virtual switch ←→ VM virtual NIC)
- Multi-queue support in vHost-user
- Tunneling support : VxLAN, GRE, Geneve
- QoS support
  - VLAN / MPLS
  - Ingress / egress policing
- DPDK statistics



❖ OVS-DPDK Details

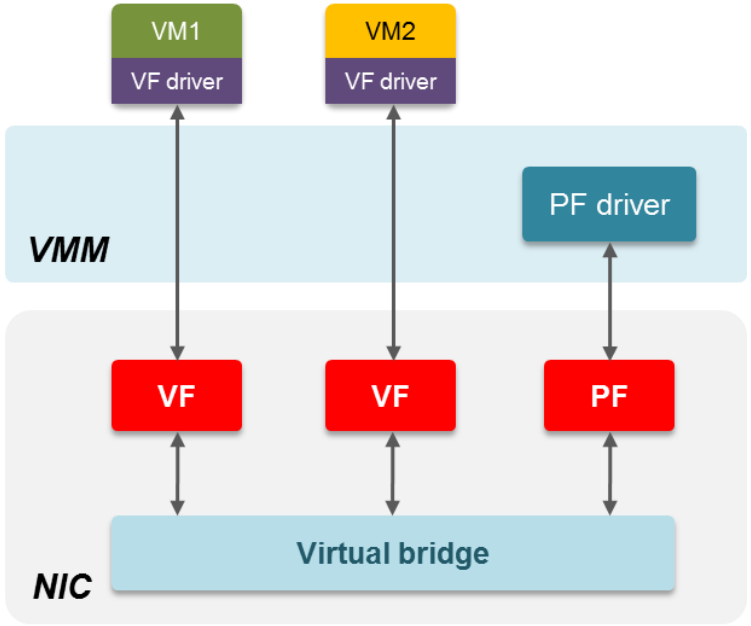
- Implements the fast path in user space using DPDK
- Consists of three types of threads
  - Packet reception
    - Gets packets from all relevant interfaces
  - Flow classification / lookup
    - Performs based on packet header information
  - Packet transmission
    - Forwards packets to the relevant interfaces
- Uses a shared memory (RING) for inter-process/thread communication



SR-IOV

❖ SR-IOV (Single Root I/O Virtualization)

- Enables VMs to share physical NIC resources
  - A NIC and its driver should provide a mechanism to segment the resources
- Virtual Function (VF)
  - Packet transmission channel to bypass the hypervisor (VM Manager, VMM)
  - Shares the common PCI resources (Physical Function, PF)
- To handle requests from multiple VFs,
  - A NIC handles single queue (Single Root) or multiple queues (Multi Root)



SR-IOV

❖ Hybrid Approach

- SR-IOV and virtual switching can be used simultaneously within a host server
  - VM 0, 1: on a datapath that bypasses the vSwitch
  - VM 2, 3: on a datapath that go through the vSwitch fabric