DPDK DATA PLANE DEVELOPMENT KIT

THE LINUX FOUNDATION

Towards Low Latency Interrupt Mode DPDK

David Su david.w.su@intel.com

Yunhong Jiang yunhong.jiang@intel.com

Wei Wang wei.w.wang@intel.com

主办方： intel

参与方： 腾讯云 ZTE 美团云 Panabit 太一星晨 Balance Your Networks UnitedStack 有云 云杉网络 Yunshan Networks

协办方： SDNLAB 专注网络创新技术 视频支持方： IT大咖说

Legal Disclaimer

- INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL® PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. INTEL PRODUCTS ARE NOT INTENDED FOR USE IN MEDICAL, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS.
- Intel may make changes to specifications and product descriptions at any time, without notice.
- All products, dates, and figures specified are preliminary based on current expectations, and are subject to change without notice.
- Intel, processors, chipsets, and desktop boards may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.
- Intel and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
- *Other names and brands may be claimed as the property of others.
-

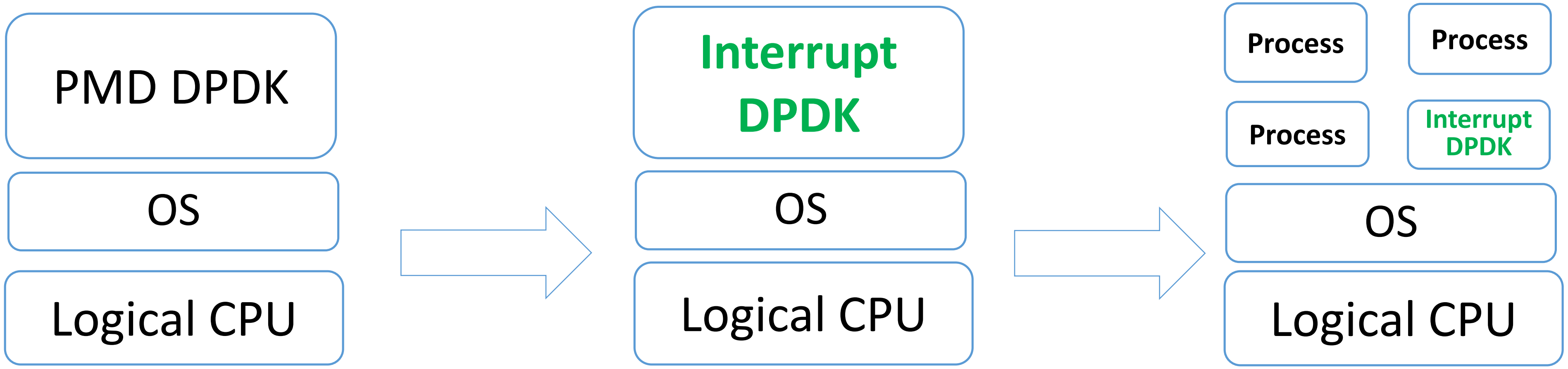
LEGAL DISCLAIMER

- No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.
- Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.
- This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.
- The products and services described may contain defects or errors known as errata which may cause deviations from published specifications. Current characterized errata are available on request.
- Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or by visiting: http://www.intel.com/design/literature.htm
- Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.
- *Other names and brands may be claimed as the property of others.
- Copyright © 2017, Intel Corporation. All rights reserved.
- Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice. Notice Revision #20110804

Agenda

- DPDK Working Mode Transition
- Problems and Optimizations
- Performance Evaluation
- Next Step Plan

Working Model Transition

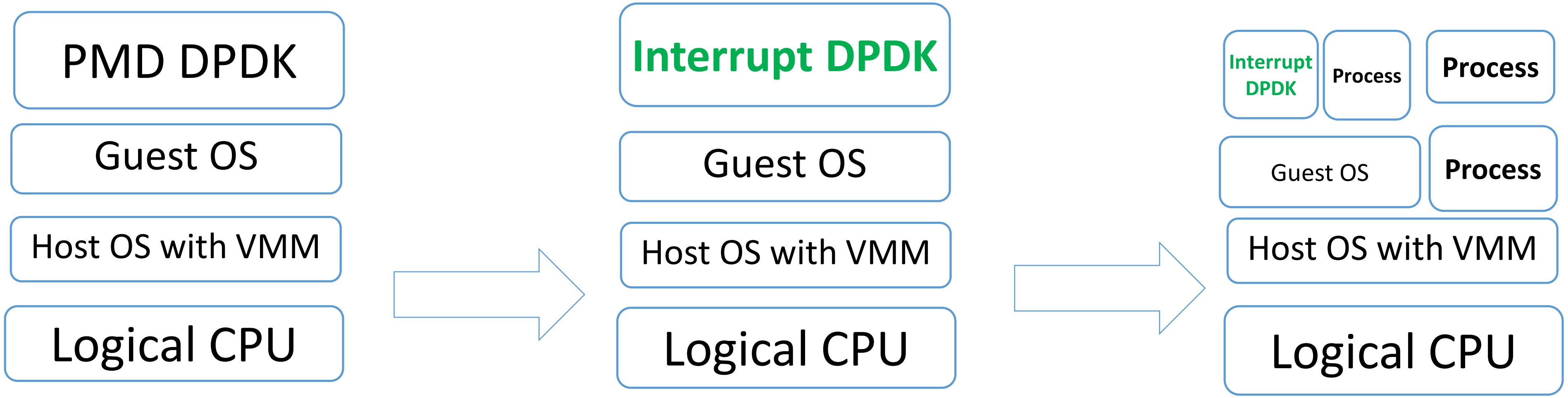


- Polling mode:
  - 100% CPU usage even without inbound packets

- Interrupt mode DPDK on a dedicated CPU:
  - Enter CPU idle state when no packet is received

- Interrupt mode DPDK sharing a CPU with other processes:
  - Run with the highest priority
  - Yield the CPU to other processes when no packet is received

Working Model Transition with Virtualization



- Polling mode:
  - 100% CPU usage even without inbound packets

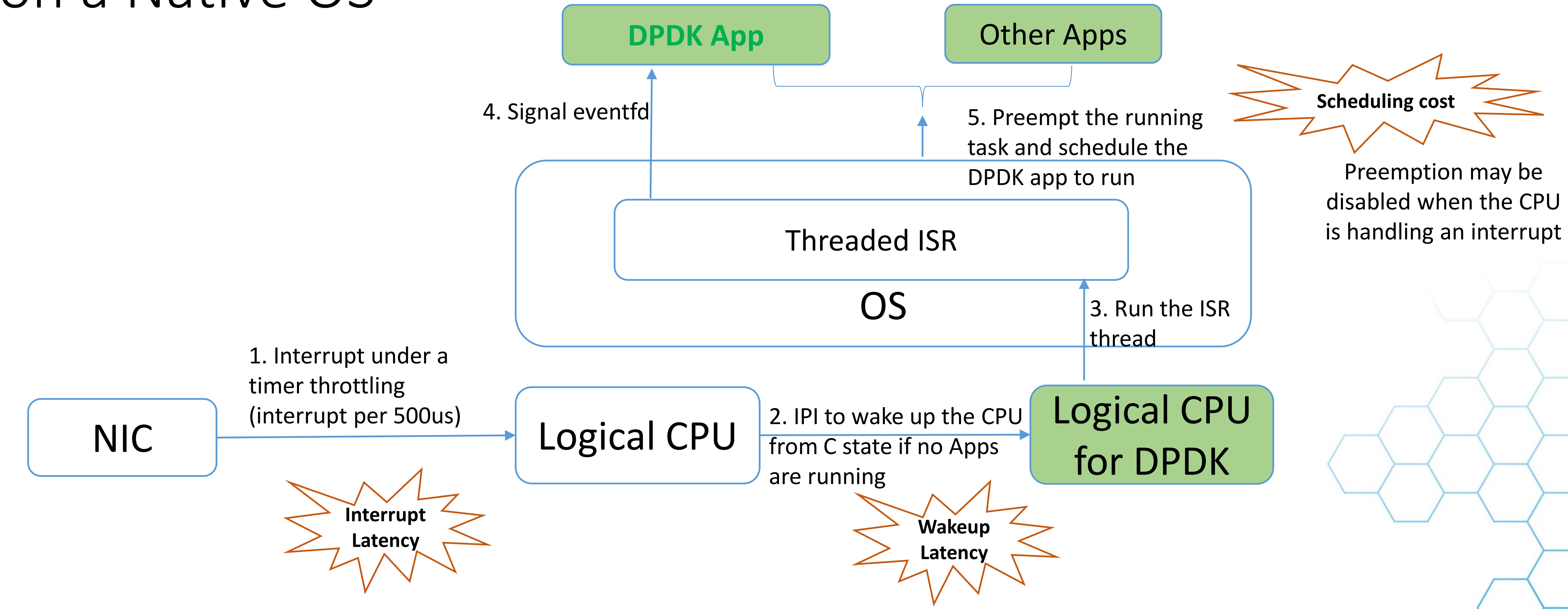
- Interrupt mode DPDK inside a VM on a dedicated CPU:
  - Enter CPU idle state when no inbound packets

- Interrupt mode DPDK inside a VM sharing a CPU with other processes:
  - Run with the highest priority
  - Yield the CPU to other processes on the Host OS when no inbound packets
  - Possible to share the CPU with processes inside the VM, but not encouraged currently.

Agenda

- DPDK Working Mode Transition
- Problems and Optimizations
- Performance Evaluation
- Next Step Plan

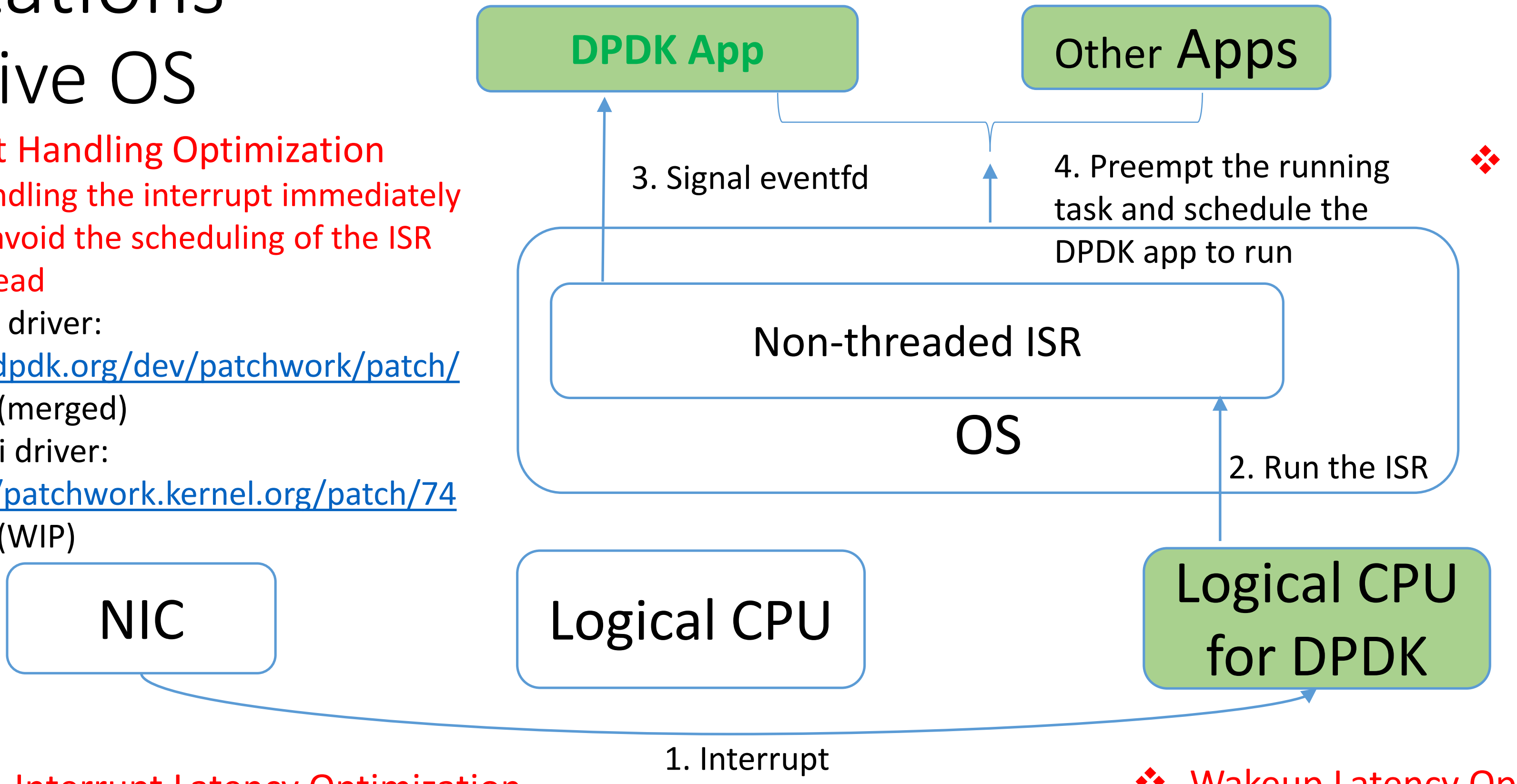
Performance Issues on a Native OS


DPDK App
Other Apps
4. Signal eventfd
5. Preempt the running task and schedule the DPDK app to run
Scheduling cost
Preemption may be disabled when the CPU is handling an interrupt
Threaded ISR
OS
3. Run the ISR thread
1. Interrupt under a timer throttling (interrupt per 500us)
NIC
Logical CPU
2. IPI to wake up the CPU from C state if no Apps are running
Logical CPU for DPDK
Interrupt Latency
Wakeup Latency

Optimizations on a Native OS

- Interrupt Handling Optimization
  - Handling the interrupt immediately to avoid the scheduling of the ISR thread

  igb_uio driver: http://dpdk.org/dev/patchwork/patch/19855/(merged)
  vfio_pci driver: https://patchwork.kernel.org/patch/7493081/(WIP)



- Scheduling Optimization
  - RT Linux is helpful to reduce the scheduling delay

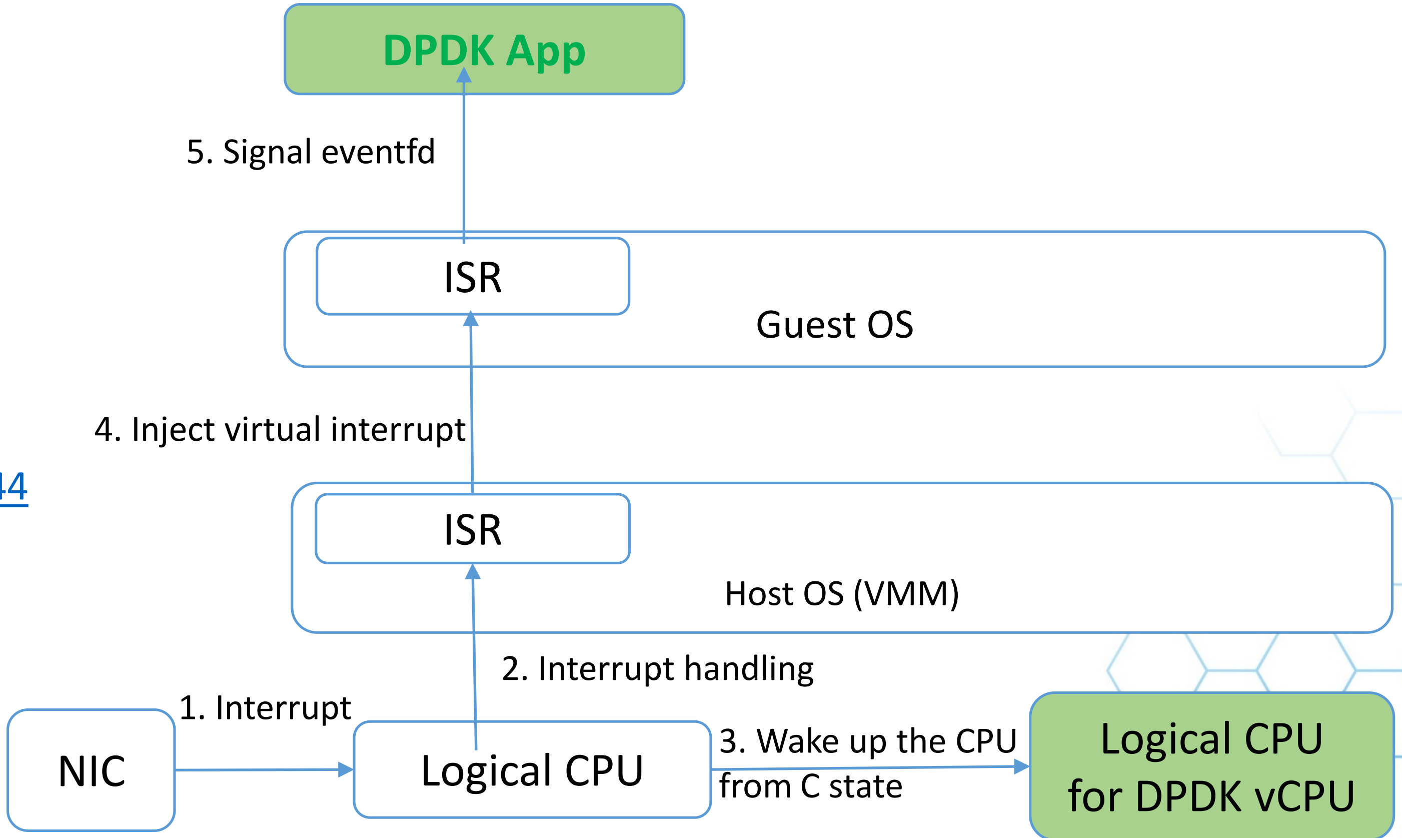
- Interrupt Latency Optimization
  - Interrupt affinity setup to avoid one IPI. It will be good if the affinity can be set in the DPDK library.
  - Remove the timer throttling to get interrupts in time. http://dpdk.org/dev/patchwork/patch/19856/ (WIP)

- Wakeup Latency Optimization
  - Limit the maximum C state via the kernel booting parameter

Performance Issues on a VM

- Latency as described for the native environment, plus the extra latency from the virtualization layer
  - The ISR on the guest kernel
  - Host/Guest context switch for interrupt injection
- Potential bugs on the VMM layer may cause longer latency
  - https://www.spinics.net/lists/kvm/msg144469.html

Further optimizations to the VMM layer are in our next step plan

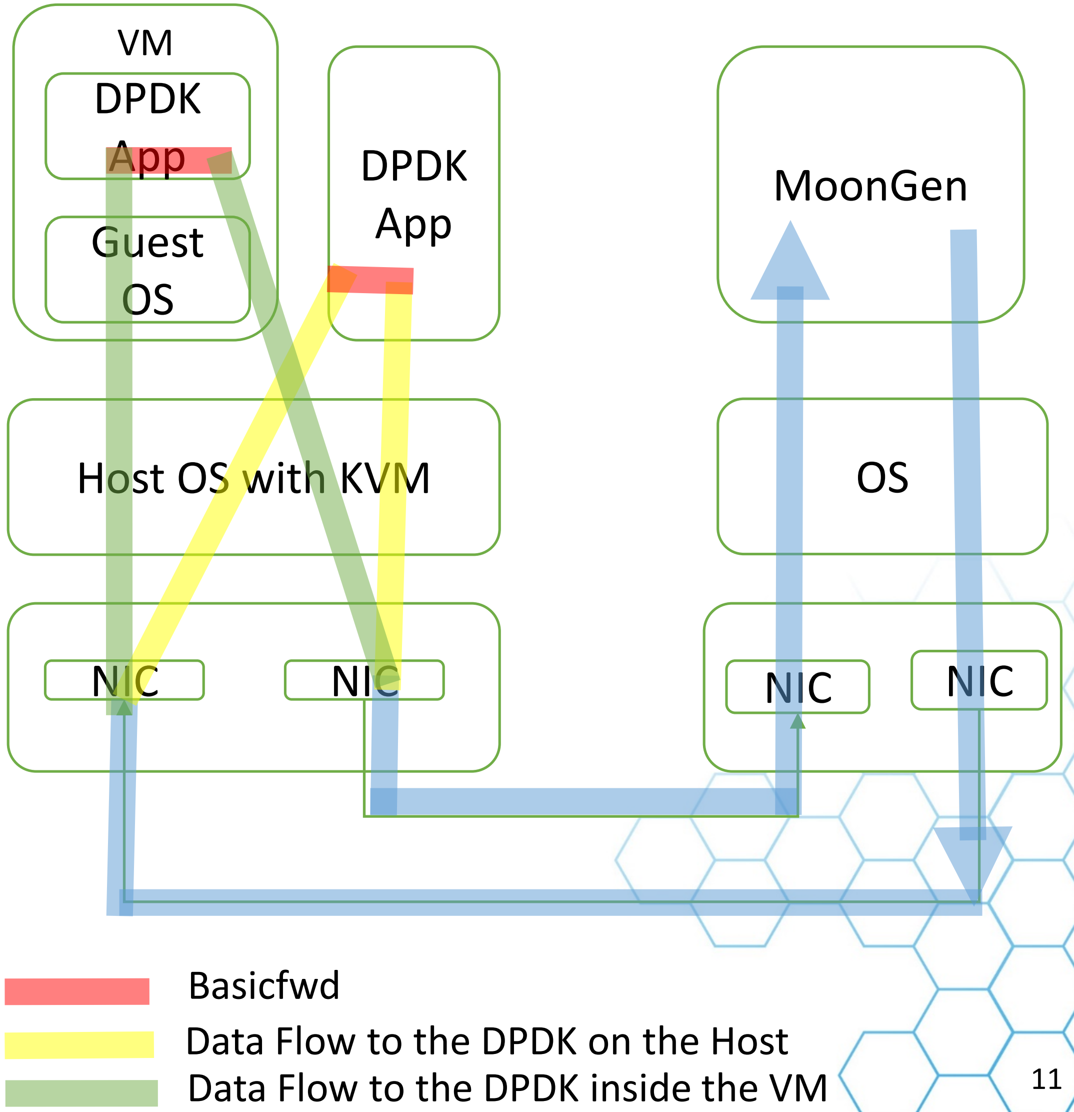


Agenda

- DPDK Working Mode Transition
- Problems and Optimizations
- Performance Evaluation
- Next Step Plan

Test Environment

- Host
  - CPU: Intel XeonE5-2699 v3 @ 2.30GHz
  - OS: KVM4NFV D release (RT Kernel 4.4)
  - NIC: Intel Ethernet Controller X540-AT2, 10Gbs
- Guest
  - vCPUs bound to isolated pCPUs
  - OS: same as host
- Test applications
  - DPDK basicfwd
    - Modified based on DPDK l3fwd-power example
    - Sleep if no packets for more than 300 us
- Packet generator (MoonGen)
  - 1 packet every 350 us



VM

DPDK App

Guest OS

DPDK App

MoonGen

Host OS with KVM

OS

NIC

NIC

NIC

NIC

🟥 Basicfwd

🟨 Data Flow to the DPDK on the Host

🟩 Data Flow to the DPDK inside the VM

CPU Idle Optimization –Current Situation

Max Cstate	C0	C1	C3	C6
Interrupt mode Basicfwd Latency (us)	14	14.9	60.9	87.7
C State Exit Latency *	0	2	33	133

* Output from “cpupower idle-info” on Intel XeonE5-2699 v3 @ 2.30GHz

Latency Improvement

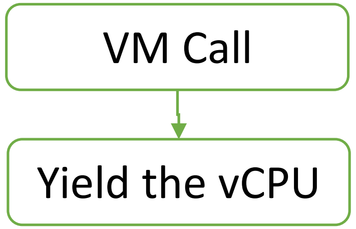
Latency	Minimum (µs)	Average (µs)	Maximum (µs)
Interrupt mode Basicfwd (Host, before optimization)	19	105	418
Interrupt mode Basicfwd (Host, after optimization)	9	14	21
Interrupt mode Basicfwd (in-VM, before optimization)	9	112	7210
Interrupt mode Basicfwd (in-VM, after optimization)	9	20	35

Agenda

- DPDK Working Mode Transition
- Problems and Optimizations
- Performance Test
- Next Step Plan

Optimizations to DPDK inside the VM

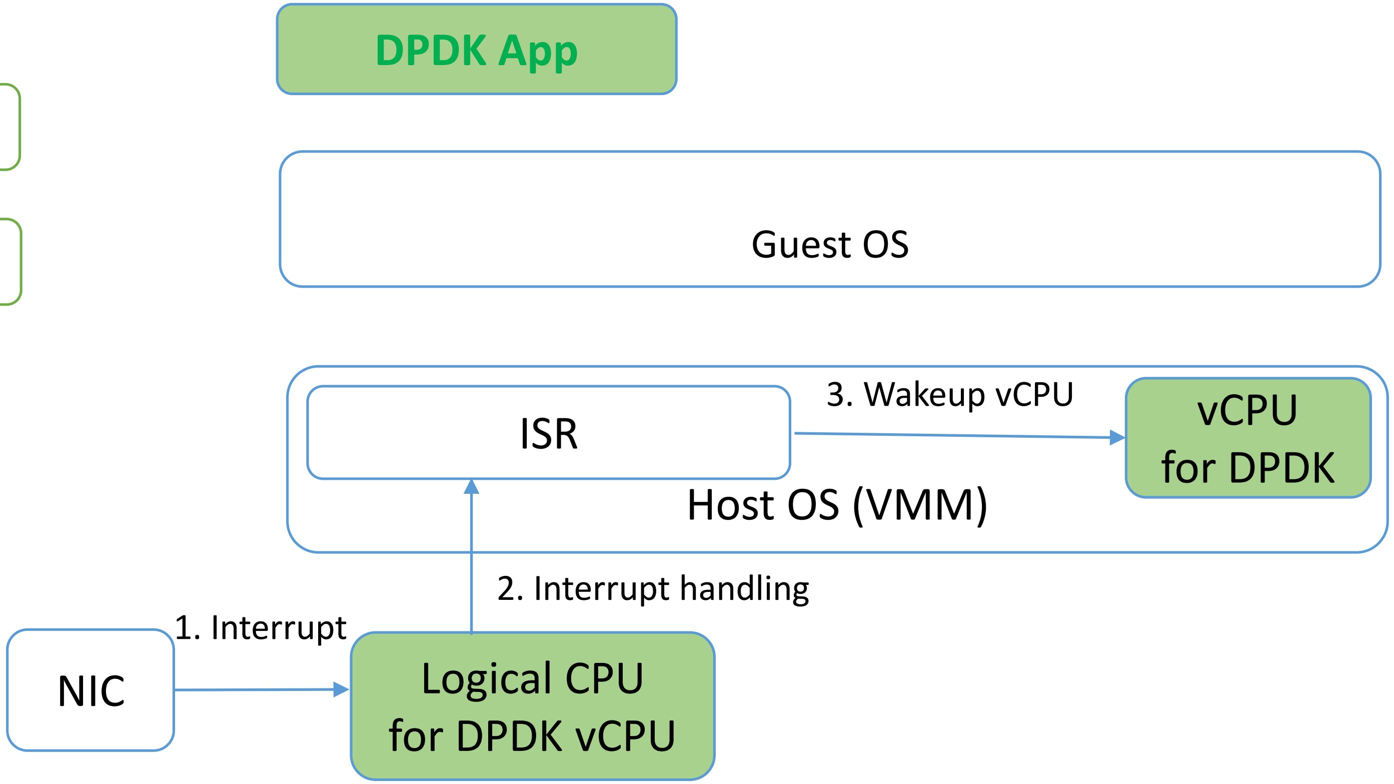
When no packets come:



VM Call

Yield the vCPU

When packets come:



DPDK App

Guest OS

ISR

3. Wakeup vCPU

vCPU for DPDK

Host OS (VMM)

2. Interrupt handling

1. Interrupt

NIC

Logical CPU for DPDK vCPU

- DPDK App starts to run once the vCPU is woken up by the Host ISR
- No need to inject virtual interrupts
- No need to signal eventfd inside the VM

End of Presentation

Thank you!