DPDK
DATA PLANE DEVELOPMENT KIT

Revise 4K Pages Performance Impact For DPDK Applications

LEI YAO
JIAYU HU

Agenda

- Why We Try To enable 4K-Page in DPDK
- The Performance Bottleneck using 4K-Page Memory
- Using 4K-Page Memory With Different Workload
- Improve the Performance

Benefit Using 4K Page Memory

- Flexible: 4K page can be allocated on demand, no memory will be preserved when system start up
- Security: No root permission need, it brings more security assurance

Why 4K page is rarely used in DPDK?

Performance Concern!

How To Use Pure 4K Page in DPDK

From DPDK 17.11 release, 4K page is supported in DPDK with VFIO-PCI driver. IOMMU can help for the IOVA to PA translation

1. **Turn on VT-d in BIOS:**

2. **Turn off transparent hugepage and enable IOMMU in Grub**

```
transparent_hugepage=never,  intel_iommu=on
```

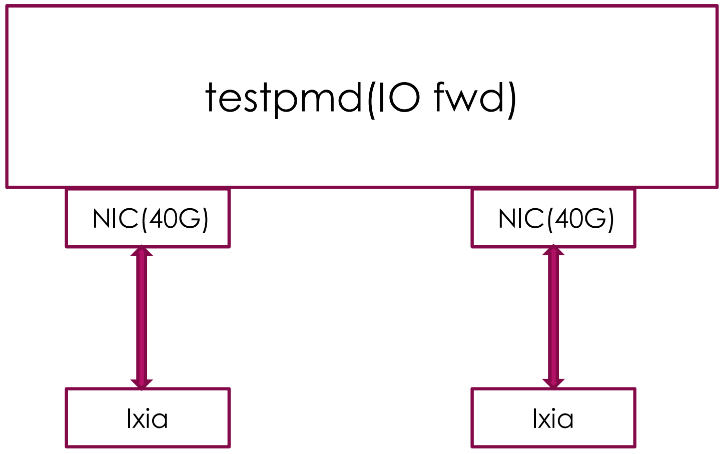
3. **Bind all NIC use in DPDK to vfio-pci, then launch DPDK application**

```
usertools/dpdk-devbind.py -b vfio-pci [BDF]
```

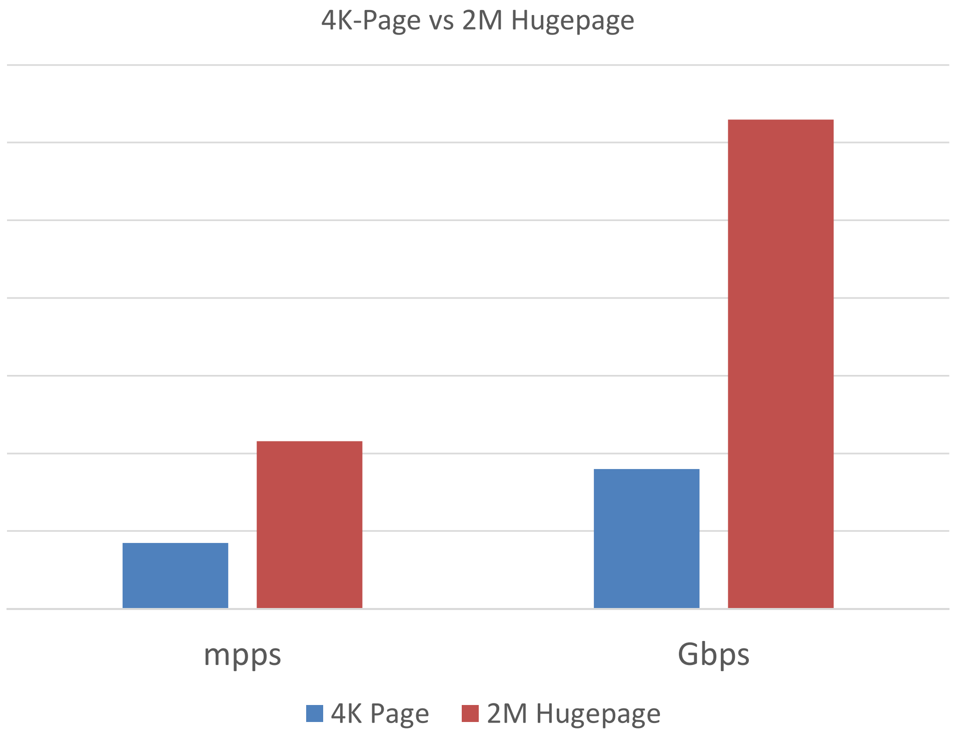
DPDK will use IOVA as VA mode in this setting, then the 4K based IOVA address will be translated to PA by Intel IOMMU, Sample command:

```
testpmd -l 1-3 -n 4 -m 1024 --no-huge  -- -i
```

First Impression Using 4K Page in DPDK

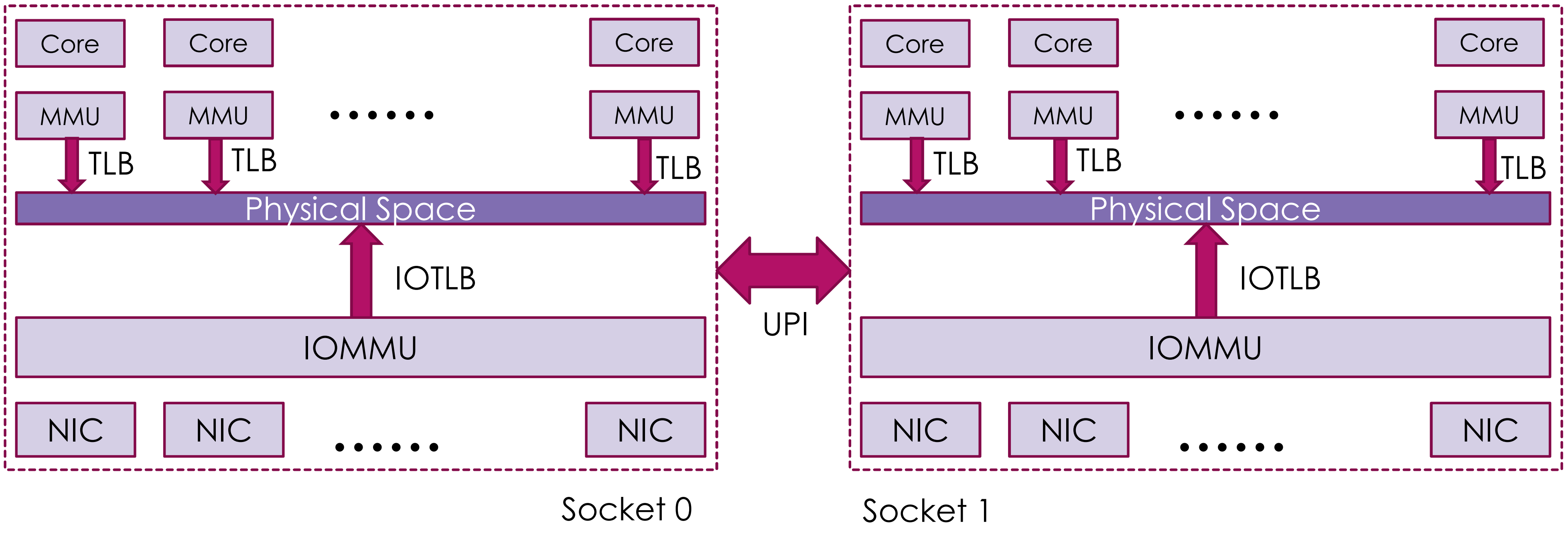


IMIX packets (64B:7, 570:4,1518:1) IMIX packets (64B:7, 570:4,1518:1)



Potential Performance Bottleneck

Potential Performance Bottleneck: TLB, IOTLB, UPI



The TLB? The UPI ?

The TLB? NO

Perf result show that the TLB miss event number in 4K is higher than 2M hugepage, but the total miss rate is still low.

4K Page:

```
#         time          counts          unit events
5.000105834      26,414,244      dTLB-load-misses       #   0.43% of all dTLB cache hits
5.000105834    6,213,669,079     dTLB-loads
```

2M Hugepage:

```
#         time          counts           unit events
5.000104143            209           dTLB-load-misses       #   0.00% of all dTLB cache hits
5.000104143    4,792,519,063     dTLB-loads
```

The UPI? NO

Remote Memory vs Local Memory - 3.1%

*** Result on Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz**

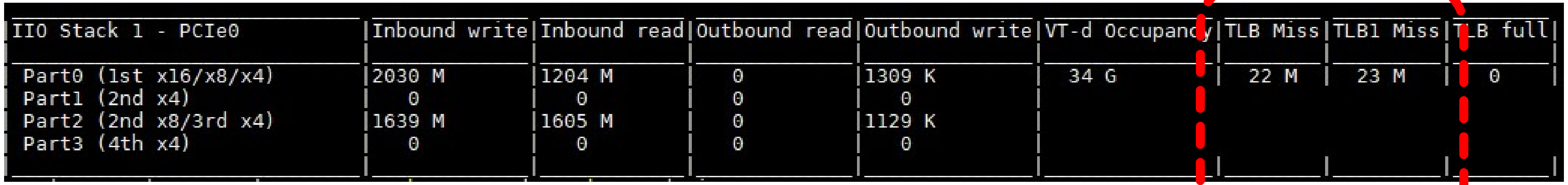
IOTLB?

Yes.

IOMMU is per socket nor per core. On Intel latest Xeon Processor , Processor Counter Monitor (PCM) can trace IOTLB miss event.

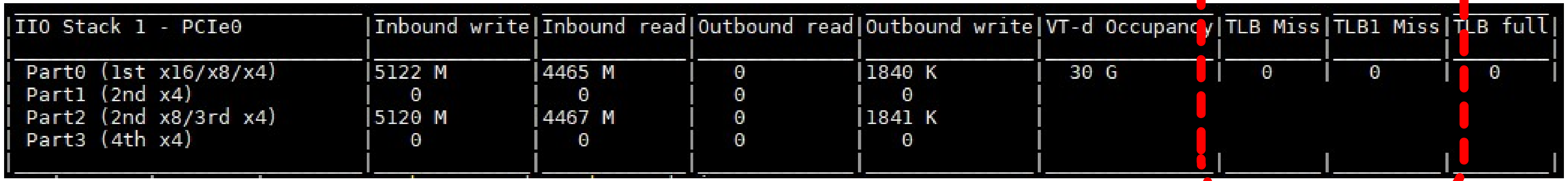
/pcm/pcm-iio.x:

4K Page:



```
|IIO Stack 1 - PCIe0                |Inbound write|Inbound read|Outbound read|Outbound write|VT-d Occupancy|TLB Miss|TLB1 Miss|TLB full|
|                                   |             |            |             |              |              |        |         |        |
| Part0 (1st x16/x8/x4)             |2030 M       |1204 M      |    0        |1309 K        |  34 G        |  22 M  |  23 M   |   0    |
| Part1 (2nd x4)                    |   0         |   0        |    0        |   0          |              |        |         |        |
| Part2 (2nd x8/3rd x4)             |1639 M       |1605 M      |    0        |1129 K        |              |        |         |        |
| Part3 (4th x4)                    |   0         |   0        |    0        |   0          |              |        |         |        |
|___________________________________|_____________|____________|_____________|______________|______________|________|_________|________|
```

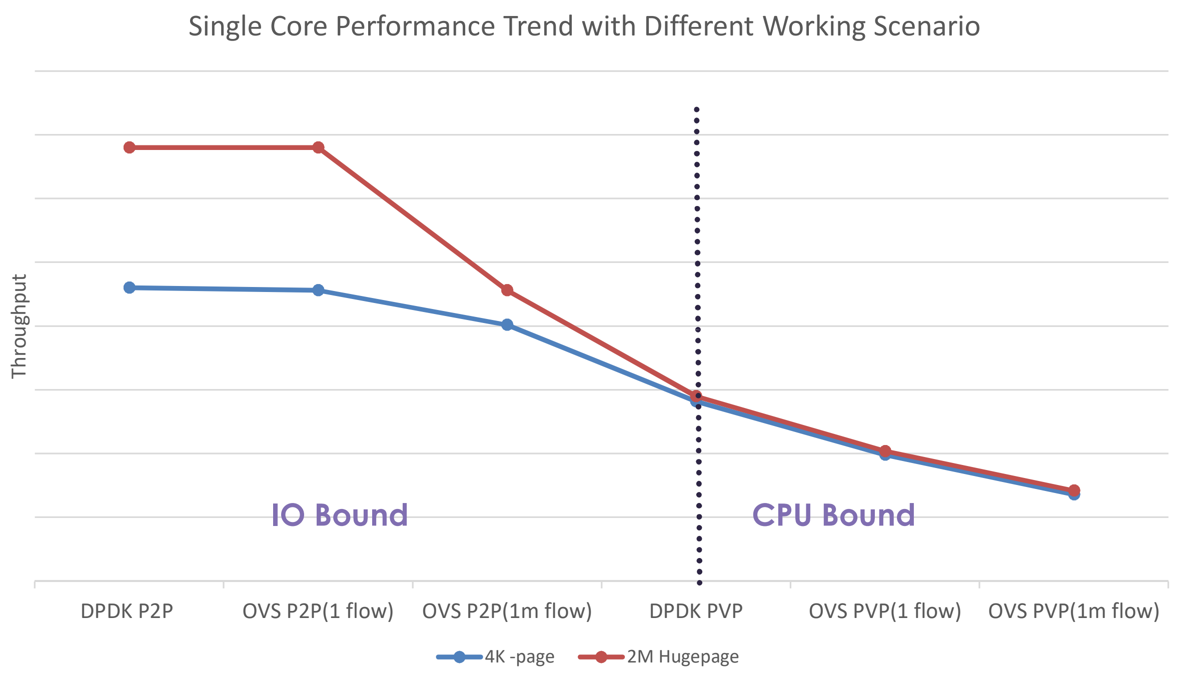
2M Hugepage



```
|IIO Stack 1 - PCIe0                |Inbound write|Inbound read|Outbound read|Outbound write|VT-d Occupancy|TLB Miss|TLB1 Miss|TLB full|
|                                   |             |            |             |              |              |        |         |        |
| Part0 (1st x16/x8/x4)             |5122 M       |4465 M      |    0        |1840 K        |  30 G        |   0    |   0     |   0    |
| Part1 (2nd x4)                    |   0         |   0        |    0        |   0          |              |        |         |        |
| Part2 (2nd x8/3rd x4)             |5120 M       |4467 M      |    0        |1841 K        |              |        |         |        |
| Part3 (4th x4)                    |   0         |   0        |    0        |   0          |              |        |         |        |
|___________________________________|_____________|____________|_____________|______________|______________|________|_________|________|
```

*** Result on Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz**

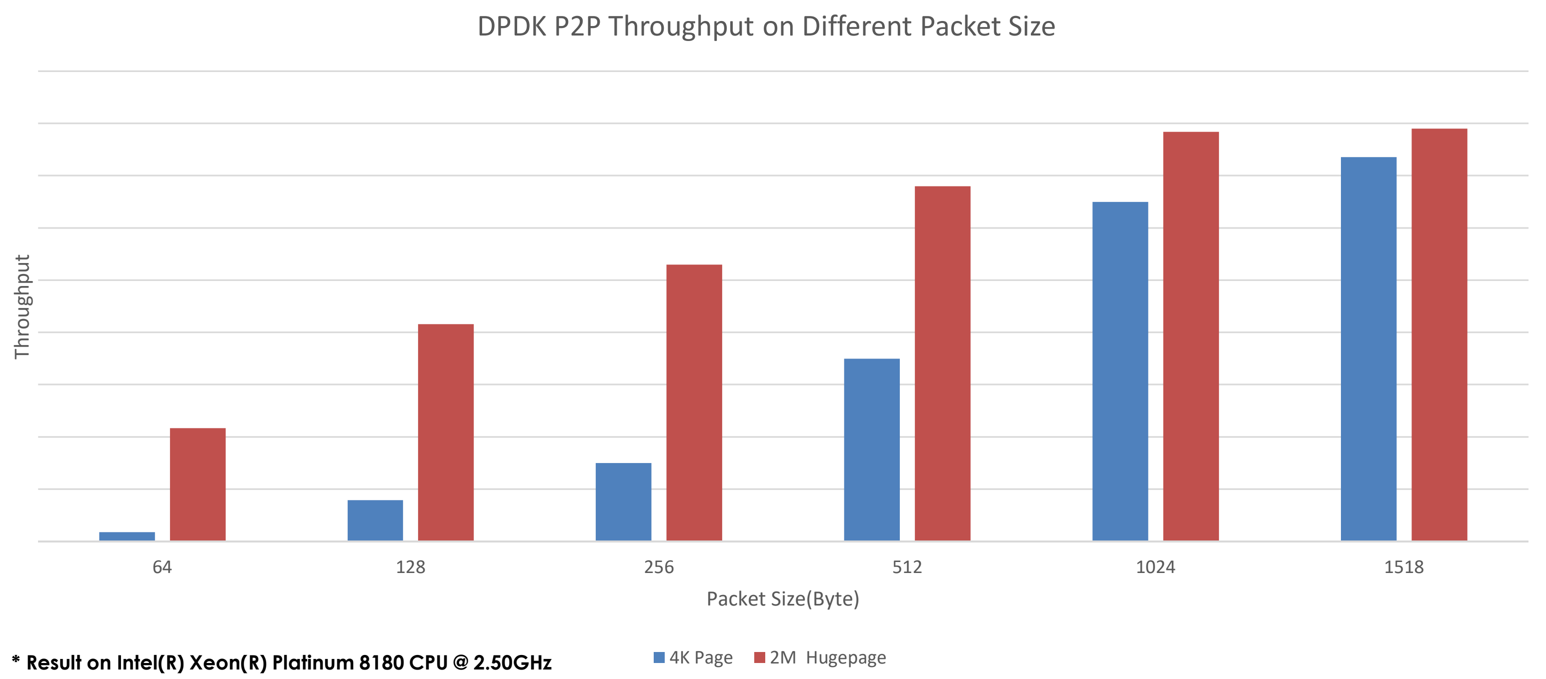
Is IOMMU Bottleneck For All Scenario?



*** Result on Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz**

4K-page memory is not the bottleneck for most of the workload in real world.

Is 4K-Page Lack of Performance at All Packet Size ?



*** Result on Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz**

The key performance gap comparing from 4K Page Memory is between packet size 64B ~ 256B

How to Improve IO Performance for Small Packets

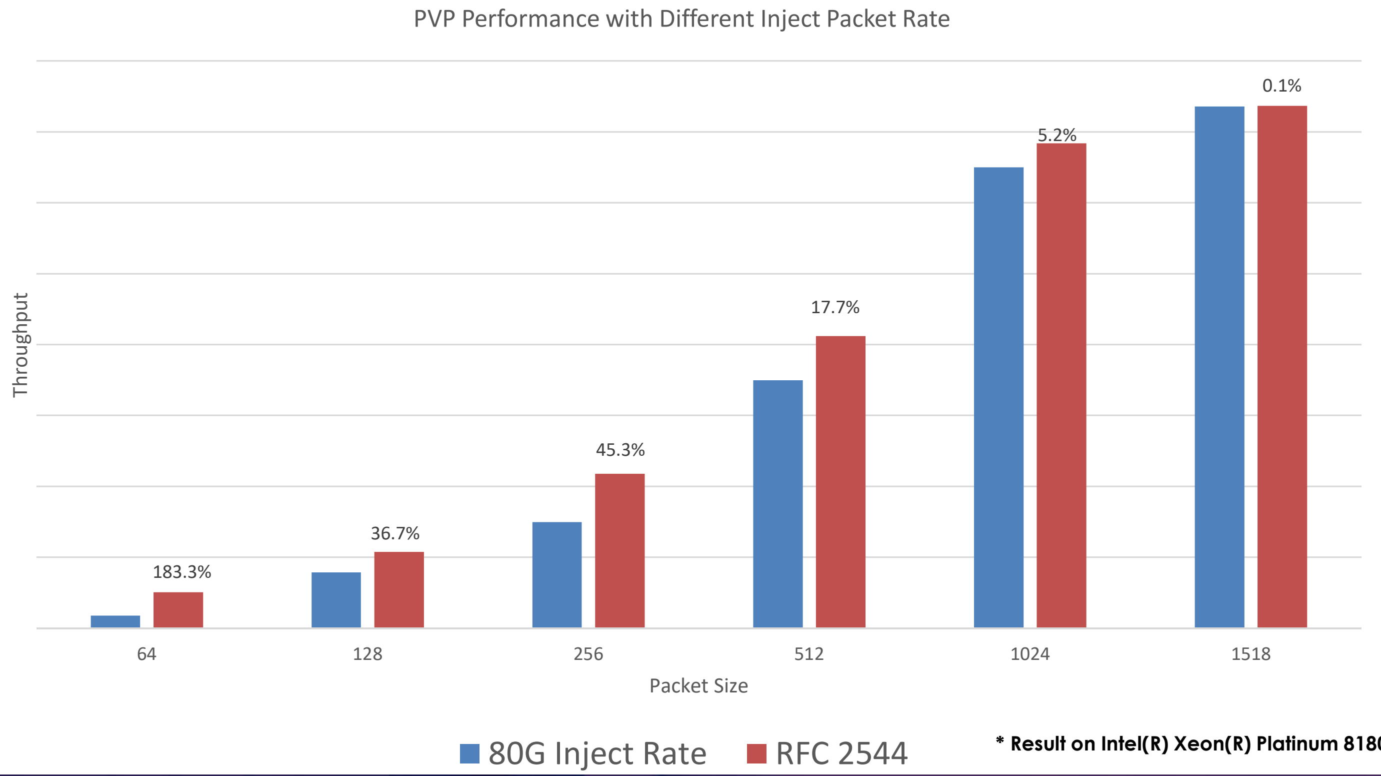
Reduce NIC txd/rxd to avoid too much IOTLB miss

IMIX Packets:	
Performance of 4K-page with rxd/txd = 1024 :	
Performance of 4K-page with rxd/txd = 128:	+ 50%
64B Packets:	
Performance of 4K-page with rxd/txd = 1024 :	
Performance of 4K-page with rxd/txd = 128:	+ 123%
128B Packets:	
Performance of 4K-page with rxd/txd = 1024 :	
Performance of 4K-page with rxd/txd = 128:	+ 9%
256B Packets:	
Performance of 4K-page with rxd/txd = 1024 :	
Performance of 4K-page with rxd/txd = 128:	+ 10%

It can work but not good idea. This method has some limitation.

How to Improve IO Performance for Small Packets

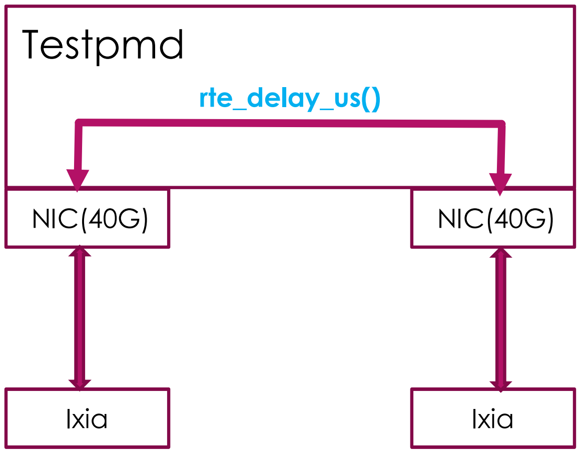
Limited the RX rate!
Option 1: Limit the Inject Packet Rate

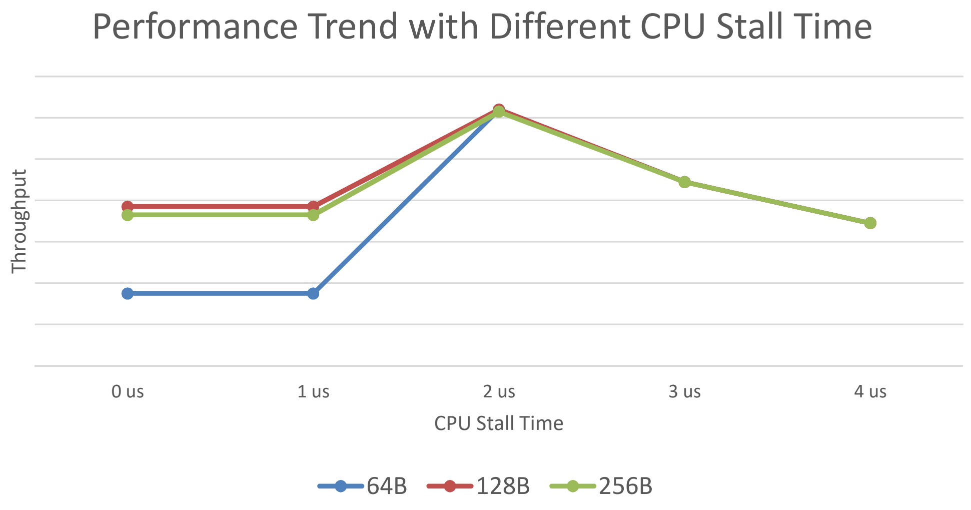


* Result on Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz

How to Improve IO Performance for Small Packets

Option 2. Slow down the frequency touching the NIC Rxd





*** Result on Intel(R) Xeon(R) Platinum 8180 CPU @ 2.50GHz**

On my test bench, 2us is the best practice number to for IO fwd workload.

Can Transparent Hugepage help?

- No Performance Gain for IMIX Packets
- No IOTLB miss event improvement.
- Default is “madvise” in latest Kernel, not always

Conclusion

- Only in IO bound scenario, 4K-page has performance concern
- IOMMU is the performance bottleneck
- Reduce the RX rate can be helpful for system throughput
- Transparent hugepage can’t help throughput

DPDK
DATA PLANE DEVELOPMENT KIT

Thank You !