INTEL® XEON® E5-2680 PROCESSOR (DP)/INTEL® C602 CHIPSET (INTEL® BD82C602 PCH): INTEL® DPDK VSWITCH PERFORMANCE REPORT

18/03/2014
LEGAL DISCLAIMERS

By using this document, in addition to any agreements you have with Intel, you accept the terms set forth below.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

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.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Optimization Notice

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

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document 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.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm
LEGAL DISCLAIMERS

All dates specified are target dates, are provided for planning purposes only and are subject to change.

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. Source: Intel internal testing as of <#TODO>, 2014. For more information go to http://www.intel.com/performance

Requires a system with a 64-bit enabled processor, chipset, BIOS and software. Performance will vary depending on the specific hardware and software you use. Consult your PC manufacturer for more information. For more information, visit http://www.intel.com/info/em64t

Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. Go to: http://www.intel.com/products/processor_number

*Other names and brands may be claimed as the property of others. Intel, the Intel logo, Xeon and SpeedStep are trademarks or registered trademarks of Intel Corporation in the U.S. and/or other countries.

Copyright © Intel Corporation 2014. All rights reserved.
PLATFORM
Intel® Server System P4308CP4MHGC Overview

- Dual Intel® Xeon® E5 family Processors
- Memory and I/O Controller integrated in CPU on a single die.
- 6 x8 electrical PCIe Gen3 IO slots

**Memory**
- DDR3 1333/1600 RDIMM with ECC
- Supports 16 DIMMs
CONFIGURATION
Configuration: Platform (DUT)

- Intel® Server System P4308CP4MHGC
- Intel® Xeon® E5-2680 processors, 2.70GHz, 20MB L3 cache
- Dual-Processor configuration
- Intel® C602 Chipset
- DDR3 1333MHz, 8 x dual rank registered ECC 4GB (total 32GB), 4 memory channels per socket Configuration, 1 DIMM per channel
- 1 x Intel® 82599 dual-port PCI-E Gen2 x8 10 Gb Ethernet NIC
Configuration: Software

• Operating system: Fedora* Core 16
• Kernel version: 3.1.0-7.fc16.x86_64
• IxExplorer*: 6.30.851.7 EA SP1
• Intel® DPDK: DPDK-1.6.0-18
Configuration: Linux* Kernel

Add the following to the kernel boot parameters and regenerate grub.conf:

default_hugepagesz=1G hugepagesz=1G hugepages=8 'intel_iommu=off' isolcpus=1[,...]
## Configuration: BIOS

<table>
<thead>
<tr>
<th>Feature</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enhanced Intel SpeedStep®</td>
<td>DISABLED</td>
</tr>
<tr>
<td>Processor C3</td>
<td>DISABLED</td>
</tr>
<tr>
<td>Processor C6</td>
<td>DISABLED</td>
</tr>
<tr>
<td>Intel® Hyper-Threading Technology (HTT)</td>
<td>DISABLED</td>
</tr>
<tr>
<td>Intel® Virtualization Technology for Directed I/O (VT-d)</td>
<td>DISABLED</td>
</tr>
<tr>
<td>MLC Streamer</td>
<td>ENABLED</td>
</tr>
<tr>
<td>MLC Spatial Prefetcher</td>
<td>ENABLED</td>
</tr>
<tr>
<td>DCU Data Prefetcher</td>
<td>ENABLED</td>
</tr>
<tr>
<td>DCU Instruction Prefetcher</td>
<td>ENABLED</td>
</tr>
<tr>
<td>Direct Cache Access (DCA)</td>
<td>ENABLED</td>
</tr>
<tr>
<td>CPU Power and Performance Policy</td>
<td>Performance</td>
</tr>
<tr>
<td>Memory Power Optimization</td>
<td>Performance Optimized</td>
</tr>
<tr>
<td>Intel® Turbo boost</td>
<td>Off</td>
</tr>
<tr>
<td>Memory RAS and Performance Configuration -&gt; NUMA</td>
<td>ENABLED</td>
</tr>
<tr>
<td>Intel®® Turbo boost</td>
<td>Off</td>
</tr>
<tr>
<td>Memory RAS and Performance Configuration -&gt; NUMA</td>
<td>ENABLED</td>
</tr>
<tr>
<td>Intel®® Turbo boost</td>
<td>Off</td>
</tr>
<tr>
<td>Memory RAS and Performance Configuration -&gt; NUMA</td>
<td>ENABLED</td>
</tr>
</tbody>
</table>
## Configuration: Host Core Affinity

### Process | Core | Comments
--- | --- | ---
client_switching_core | 1 | Affinity Set in `ovs_dpdk` command line
RX Core Port 0 | 2 | Affinity Set in `ovs_dpdk` command line
RX Core Port 1 | 3 | Affinity Set in `ovs_dpdk` command line
vswitchd | 8 | `taskset -a <pid_of_vswitchd_process>`

### Switching Cores

#### QEMU* VM1 VCPU0
- `taskset -c -p 4 <pid_of_vm1_qemu_vcpu0_process>`

#### QEMU* VM1 VCPU1
- `taskset -c -p 5 <pid_of_vm1_qemu_vcpu1_process>`
- `taskset -a -c -p 5 <pid_of_qemu_vm1_process>`

### OS/VM Cores

#### QEMU* VM2 VCPU0
- `taskset -c -p 6 <pid_of_vm2_qemu_vcpu0_process>`

#### QEMU* VM2 VCPU1
- `taskset -c -p 7 <pid_of_vm2_qemu_vcpu1_process>`
- `taskset -a -c -p 7 <pid_of_qemu_vm2_process>`

#### Kernel
- 0 | All other CPUs isolated (`isolcpus` boot parameter)

---

[1] QEMU* VM affinity not required for Phy-Phy tests
[2] QEMU* VM2 affinity not required for loopback tests, assuming VM1 is used for testing
Configuration: PHY-PHY Test Architecture

Hardware View

Host Machine

Eight Core Intel® Xeon® E5-2680 Processor

10GbE ports

PCI-E Gen3 x8

Ixia* 10 Gigabit Ethernet Traffic Generator

Software View

Host Machine

Intel® DPDK vSwitch

Phy Port 1

Phy Port 2

Network
Configuration: VM Loopback Test Architecture

Hardware View

Host Machine

- Eight Core Intel® Xeon® E5-2680 Processor
- 10GbE ports
- PCI-E Gen3 x8

Ixia* 10 Gigabit Ethernet Traffic Generator

Software View

Host Machine

- VM
- Intel® DPDK vSwitch

Network

Phy Port 0

Phy Port 1
Configuration: VM-VM Test Architecture

**Hardware View**
- **Host Machine**
  - Eight Core Intel® Xeon® E5-2680 Processor
  - 10GbE ports
  - PCI-E Gen3 x8
- Ixia* 10 Gigabit Ethernet Traffic Generator

**Software View**
- **Host Machine**
  - Intel® DPDK vSwitch
- **Network**
  - Phy Port 0
  - Phy Port 1
  - VM
  - VM
Configuration: General Test Setup

```bash
sudo mkdir -p /dev/hugepages
sudo mount -t hugetlbfs nodev /dev/hugepages
sudo /sbin/rmmod ixgbe
sudo /sbin/modprobe uio
sudo /sbin/insmod $DPDK/$DPDK_TARGET/kmod/igb_uio.ko

echo 0 > /proc/sys/kernel/randomize_va_space
```
Configuration: IVSHM Host Common

```bash
sudo rm -rf <QEMU_SHARE_DIR>
sudo mkdir -p <QEMU_SHARE_DIR>
chmod 777 <QEMU_SHARE_DIR>
mkdir -p <QEMU_SHARE_DIR>/DPDK
mkdir -p <QEMU_SHARE_DIR>/ovs_client
cp -a <DPDK_DIR>/* <QEMU_SHARE_DIR>/DPDK
cp -aL <OVDK_DIR>/guest/ovs_client/* <QEMU_SHARE_DIR>/ovs_client
```
mkdir -p /mnt/ovs_client
mount -o iocharset=utf8 /dev/sdb1 /mnt/ovs_client
mkdir -p /root/ovs_client
cp -a /mnt/ovs_client/* /root/ovs_client
cd /root/ovs_client/DPDK

export CC=gcc
make uninstall
make install T=x86_64-ivshmem-linuxapp-gcc
export RTE_SDK=/root/ovs_client/DPDK
export RTE_TARGET=x86_64-ivshmem-linuxapp-gcc
cd /root/ovs_client

cd ovs_client
make clean
make
Configuration: KNI Host Common

rm -rf <QEMU_SHARE_DIR>
mkdir -p <QEMU_SHARE_DIR>
chmod 777 <QEMU_SHARE_DIR>
mkdir -p <QEMU_SHARE_DIR>/DPDK
mkdir -p <QEMU_SHARE_DIR>/kni_client

cp -a <DPDK_DIR>/*/ <QEMU_SHARE_DIR>/DPDK

cp -aL <OVDK_DIR>/guest/kni/* <QEMU_SHARE_DIR>
cp -aL <OVDK_DIR>/guest/kni_client/* <QEMU_SHARE_DIR>
mkdir -p /mnt/ovdk
mount -o iocharset=utf8 /dev/sdb1 /mnt/ovdk
mkdir -p /root/ovdk
cp -a /mnt/ovdk/* /root/ovdk
cd /root/ovdk/DPDK

export CC=gcc
make uninstall
patch -N -p1 < rte_kni_module_1_6.patch
export RTE_SDK=/root/kni_client/DPDK
export RTE_TARGET=x86_64-ivshmem-linuxapp-gcc
make install T=x86_64-ivshmem-linuxapp-gcc
sudo insmod /root/ovdk/DPDK/x86_64-ivshmem-linuxapp-gcc/kmod/rte_kni.ko

cd /root/ovdk/kni_client/libvport
make clean
make
cd /root/ovdk/kni_client
make clean
make
Configuration: US-Vhost Host Common

```
rm -rf <QEMU_SHARE_DIR>
mkdir -p <QEMU_SHARE_DIR>
chmod 777 <QEMU_SHARE_DIR>
mkdir -p <QEMU_SHARE_DIR>/DPDK
cp -a <DPDK_DIR>/* <QEMU_SHARE_DIR>/DPDK

sudo /sbin/insmod openvswitch/datapath/dpdk/fd_link/fd_link.ko
sudo /sbin/modprobe cuse
sudo /sbin/rmmod vhost_net
sudo rm -f /dev/vhost-net
```
RESULTS
RESULTS
Intel® Xeon® E5-2680 Processor (DP)/Intel® C602 Chipset (Intel® BD82C602 PCH): IPv4 switching with 13-tuple lookup

Intel® DPDK vSwitch Packet Switching: Phy-Phy

![Bar Chart]

- **64** bytes: 10.99
- **256** bytes: 4.53
- **512** bytes: 2.35
- **1024** bytes: 1.197
- **1518** bytes: 0.813

**Million Packets per Second**

**Packet size in bytes**

**Date:** March 2014

**Disclaimer:** Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Source: Intel internal testing as of March, 2014. See Linux* Performance Tuning for configuration details.

Additional information can be found at [http://www.intel.com/performance](http://www.intel.com/performance).

Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.
Configurations: Phy-Phy

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section.

Initial setup, as per ‘Configuration: General Test Setup’.

cd $OPENVSWITCH_DIR
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db ./vswitchd/vswitch.ovsschema
sudo ./ovsdb/ovsdb-server --remote=punix:/usr/local/var/run/openvswitch/db.sock --remote=db:Open_vSwitch,Open_vSwitch,manager_options &

sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy16 -- set Interface ovsphy16 type=dpdk ofport_request=16
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy17 -- set Interface ovsphy17 type=dpdk ofport_request=17

sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --base-virtaddr=0x2aaa2aa000000 -- -p 0x03 --stats=0 --vswitchd=0 --client_switching_core=1 --config="(0,0,2),(1,0,3)"

sudo ./vswitchd/ovs-vswitchd -c 0x100 --proc-type=secondary &
ovs-ofctl add-flow br0 in_port=16,vlan_tci=0x3258,idle_timeout=0,action=output:17

Affinitize cores, as described in ‘Configuration: Core Affinity’
VM Loopback IVSHM

RESULTS
Intel® Xeon® E5-2680 Processor (DP)/
Intel® C602 Chipset (Intel® BD82C602 PCH):
IPv4 switching with 13-tuple lookup

**Intel® DPDK vSwitch Packet Switching:**
**VM Loopback IVSHM**

<table>
<thead>
<tr>
<th>Packet size in bytes</th>
<th>Million Packets per Second</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>10.5</td>
</tr>
<tr>
<td>256</td>
<td>4.53</td>
</tr>
<tr>
<td>512</td>
<td>2.35</td>
</tr>
<tr>
<td>1024</td>
<td>1.197</td>
</tr>
<tr>
<td>1518</td>
<td>0.813</td>
</tr>
</tbody>
</table>

Date: March 2014

Disclaimer: Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Source: Intel internal testing as of March, 2014. See Linux* Performance Tuning for configuration details.

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section. Initial setup, as per ‘Configuration: General Test Setup’, and ‘Configuration: IVSHM Host Common’.

```bash
cd $OPENVSWITCH_DIR
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db ./vswitchd/vswitch.ovsschema
sudo ./ovsdb/ovsdb-server --remote=punix:/usr/local/var/run/openvswitch/db.sock --remote=db:Open_vSwitch,Open_vSwitch,manager_options &

sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsclient1 -- set Interface ovsclient1 type=dpdkclient ofport_request=1
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy16 -- set Interface ovsphy16 type=dpdkphy ofport_request=16
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy17 -- set Interface ovsphy17 type=dpdkphy ofport_request=17

sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --base-virtaddr=0x2aaa2aa0000 -- -p 0x3 -n 2 --stats=0 --vswitchd=0 --client_switching_core=1 --config="(0,0,2),(1,0,3)"
```
Configurations: VM Loopback IVSHM (Host) (Cont.)

```bash
sudo ./vswitchd/ovs-vswitchd -c 0x100 --proc-type=secondary &

sudo ./utilities/ovs-ivshm-mngr/build/app/ovs-ivshm-mngr -c 0x1 - --proc-type=secondary -vm_1:ovsclient1

sudo ./utilities/ovs-ofctl add-flow br0
   in_port=16,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=6.6.6.2,idle_timeout=0,action=output:1

sudo ./utilities/ovs-ofctl add-flow br0
   in_port=1,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=6.6.6.2,idle_timeout=0,action=output:17
```
Configurations: VM Loopback IVSHM (Guest)

Host QEMU* command line

```bash
sudo $QEMU_DIR/x86_64-softmmu/qemu-system-x86_64 -c 0x30 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm -name "client 1" -nographic -vnc :1`cat /tmp/.ovs_ivshmem_qemu_cmdline_vm_1`" -drive file=fat:rw:$QEMU_SHARE_DIR,snapshot=off -monitor unix:$VM1MONITOR,server,nowait &
```

Guest Configuration:

Initial guest setup as per ‘Configuration: IVSHM Guest Common’, then execute:

```bash
./ovs_client -c 0x1 -n 4 -- -p ovsclient1 &
```

Affinitize cores, as described in ‘Configuration: Core Affinity’
VM-VM IVSHM

RESULTS
Intel® Xeon® E5-2680 Processor (DP)/Intel® C602 Chipset (Intel® BD82C602 PCH): IPv4 switching with 13-tuple lookup

Intel® DPDK vSwitch Packet Switching: VM-VM IVSHM

Million Packets per Second

<table>
<thead>
<tr>
<th>Packet size in bytes</th>
<th>64</th>
<th>256</th>
<th>512</th>
<th>1024</th>
<th>1518</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6.51</td>
<td>4.53</td>
<td>2.35</td>
<td>1.197</td>
<td>0.813</td>
</tr>
</tbody>
</table>

Date: March 2014

Disclaimer: Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. Source: Intel internal testing as of March, 2014. See Linux* Performance Tuning for configuration details. For more information go to [http://www.intel.com/performance](http://www.intel.com/performance).

Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.
Configurations: VM-VM IVSHM (Host)

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section.
Initial setup, as per ‘Configuration: General Test Setup’, and ‘Configuration: IVSHM Host Common’.

```bash
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db ./vswitchd/vswitch.ovsschema
sudo ./ovsdb/ovsdb-server --remote=punix:/usr/local/var/run/openvswitch/db.sock --remote=db:Open_vSwitch,Open_vSwitch,manager_options &

sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk type=dpdkclient ofport_request=1
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsclient1 -- set Interface ovsclient1 type=dpdkclient ofport_request=1
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsclient2 -- set Interface ovsclient2 type=dpdkclient ofport_request=2
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy16 -- set Interface ovsphy16 type=dpdkphy ofport_request=16
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy17 -- set Interface ovsphy17 type=dpdkphy ofport_request=17

sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --base-virtaddr=0x2aaa2aa0000 --p 0x03 -n 3 --stats=0 --vswitchd=0 --client_switching_core=1 --config="(0,0,2),(1,0,3)"
```
Configurations: VM-VM IVSHM (Host) (Contd.)

```bash
sudo ./vswitchd/ovs-vswitchd -c 0x100 --proc-type=secondary &
sudo ./utilities/ovs-ivshm-mngr/build/app/ovs-ivshm-mngr -c 0x1 --proc-type=secondary --vm_1:ovsclient1 -vm_2:ovsclient2

sudo ./utilities/ovs-ofctl add-flow br0 in_port=16,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=6.6.6.2,idle_timeout=0,action=output:1
sudo ./utilities/ovs-ofctl add-flow br0 in_port=1,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=6.6.6.2,idle_timeout=0,action=output:2
sudo ./utilities/ovs-ofctl add-flow br0 in_port=2,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=6.6.6.2,idle_timeout=0,action=output:17
```
Configurations: VM-VM IVSHM (Guest)

Host QEMU* command line VM1
sudo $QEMU_DIR/x86_64-softmmu/qemu-system-x86_64 -c 0x30 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm -name "client 1" -nographic -vnc :1`cat /tmp/.ovs_ivshmem_qemu_cmdline_vm_1`` -drive file=fat:rw:$QEMU_SHARE_DIR, snapshot=off -monitor unix:VM1MONITOR,server,nowait &

Host QEMU* command line VM2
sudo $QEMU_DIR/x86_64-softmmu/qemu-system-x86_64 -c 0xC0 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm -name "client 2" -nographic -vnc :1`cat /tmp/.ovs_ivshmem_qemu_cmdline_vm_2`` -drive file=fat:rw:$QEMU_SHARE_DIR, snapshot=off -monitor unix:VM1MONITOR,server,nowait &

Guest Configuration:

Initial guest setup as per ‘Configuration: IVSHM Guest Common’.

Guest command lines:
./ovs_client -c 0x1 -n 4 -- -p ovsclient1 & #VM1
./ovs_client -c 0x1 -n 4 -- -p ovsclient2 & #VM2

Affinitize cores, as described in ‘Configuration: Core Affinity’
RESULTS

VM Loopback KNI
Intel® Xeon® E5-2680 Processor (DP)/Intel® C602 Chipset (Intel® BD82C602 PCH): IPv4 switching with 13-tuple lookup

Intel® DPDK vSwitch Packet Switching: VM Loopback KNI

<table>
<thead>
<tr>
<th>Packet size in bytes</th>
<th>Million Packets per Second</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>1.03</td>
</tr>
<tr>
<td>256</td>
<td>1</td>
</tr>
<tr>
<td>512</td>
<td>0.97</td>
</tr>
<tr>
<td>1024</td>
<td>0.78</td>
</tr>
<tr>
<td>1518</td>
<td>0.695</td>
</tr>
</tbody>
</table>

Date: March 2014

Disclaimer: Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Source: Intel internal testing as of March, 2014. See Linux* Performance Tuning for configuration details.


Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.
Configurations: VM Loopback KNI (Host)

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section.
Initial setup, as per ‘Configuration: General Test Setup’, and ‘Configuration: KNI Host Common’.

```
cd $OPENVSWITCH_DIR
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db
   ./vswitchd/vswitch.ovsschema
sudo ./ovsdb/ovsdb-server --remote=puunix:/usr/local/var/run/openvswitch/db.sock --
   remote=db:Open_vSwitch,Open_vSwitch,manager_options &

sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy16 -- set Interface ovsphy16
   type=dpdkphy ofport_request=16 option:port=0
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy17 -- set Interface ovsphy17
   type=dpdkphy ofport_request=17 option:port=1
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovskni0 -- set Interface ovskni0
   type=dpdkkni ofport_request=32
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovskni1 -- set Interface ovskni1
   type=dpdkkni ofport_request=33

sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --base-
   virtaddr=0x2aaa2aa0000 -- -p 0x03 -n 2 -k 2 --stats=0 --vswitchd=0 --
   client_switching_core=1 --config="(0,0,2),(1,0,3)"
```
Configurations: VM Loopback KNI (Host)(Contd.)

sudo ./vswitchd/ovs-vswitchd -c 0x100 --proc-type=secondary &

sudo ./utilities/ovs-ivshm-mngr/build/app/ovs-ivshm-mngr -c 0x1 --proc-type=secondary --vm_1:ovskni0,ovskni1

sudo ./utilities/ovs-ofctl del-flows br0
sudo ./utilities/ovs-ofctl add-flow br0
    in_port=16,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 32
sudo ./utilities/ovs-ofctl add-flow br0
    in_port=33,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 17
Configurations: VM Loopback KNI (Guest)

Host QEMU* command line:
`sudo $QEMU_DIR/x86_64-softmmu/qemu-system-x86_64 -c 0x30 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm -name "client 1" -nographic -vnc :1 $(cat /tmp/ovs_ivshmem_qemu_cmdline_vm_1) -drive file=fat:rw:$QEMU_SHARE_DIR,snapshot=off -monitor unix:$VM1MONITOR,server,nowait &`

Guest Configuration:

Initial guest setup as per ‘Configuration: KNI Guest Common’.

```
./build/kni_client -c 0x1 -n 4 -- -p ovskni0 -p ovskni1 &
ifconfig vEth_ovskni0 hw ether 00:4B:4E:49:30:00
ifconfig vEth_ovskni1 hw ether 00:4B:4E:49:30:01
ifconfig vEth_ovskni0 2.2.2.1/24 up
ifconfig vEth_ovskni1 3.3.3.1/24 up
sysctl -w net.ipv4.ip_forward=1
sysctl -w net.ipv4.conf.all_rp_filter=0
sysctl -w net.ipv4.conf.vEth_ovskni0_rp_filter=0
route add default gw 3.3.3.2 vEth_ovskni1
ifconfig -a
arp -s 3.3.3.2 DE:AD:BE:EF:CA:FE
```

Affinitize host cores, as described in ‘Configuration: Core Affinity’
VM-VM KNI

RESULTS
Intel® Xeon® E5-2680 Processor (DP)/Intel® C602 Chipset (Intel® BD82C602 PCH): IPv4 switching with 13-tuple lookup

Intel® DPDK vSwitch Packet Switching: VM-VM KNI

![Graph showing packet-switching performance](image)

**Results:**
- 64 packets: 1.185 Mbps
- 256 packets: 1.14 Mbps
- 512 packets: 1.062 Mbps
- 1024 packets: 0.865 Mbps
- 1518 packets: 0.806 Mbps

**Packet size in bytes:**
64, 256, 512, 1024, 1518

**Million Packets per Second**

**Date:** March 2014

**Disclaimer:** Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Source: Intel internal testing as of March, 2014. See Linux® Performance Tuning for configuration details.


Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.
Configurations: VM-VM KNI (Host)

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section.

Initial setup, as per ‘Configuration: General Test Setup’, and ‘Configuration: KNI Host Common’.

```
$ cd $OPENVSWITCH_DIR/
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db
   ./vswitchd/vswitch.ovsschema
sudo ./ovsdb/ovsdb-server --remote=punix:/usr/local/var/run/openvswitch/db.sock --
   remote=db:Open_vSwitch,Open_vSwitch,manager_options &

sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy16 -- set Interface ovsphy16
   type=dpdkphy ofport_request=16 option:port=0
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy17 -- set Interface ovsphy17
   type=dpdkphy ofport_request=17 option:port=1
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovskni0 -- set Interface ovskni0
   type=dpdkkni ofport_request=32
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovskni1 -- set Interface ovskni1
   type=dpdkkni ofport_request=33
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovskni2 -- set Interface ovskni2
   type=dpdkkni ofport_request=34
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovskni3 -- set Interface ovskni3
   type=dpdkkni ofport_request=35
```
Configurations: VM-VM KNI (Host)(Contd.)

```
sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --base-virtaddr=0x2aaa2aa0000 -- p 0x03 -k 4 --stats=0 --vswitchd=0 --client_switching_core=1 --config="(0,0,2),(1,0,3)"
sudo ./vswitchd/ovs-vswitchd -c 0x100 --proc-type=secondary &

sudo ./utilities/ovs-ivshm-mngr/build/app/ovs-ivshm-mngr -c 0x1 --proc-type=secondary --vm_1:ovskni0,ovskni2 vm_2:ovskni1,ovskni3

sudo ./utilities/ovs-ofctl del-flows br0
sudo ./utilities/ovs-ofctl add-flow br0
  in_port=16,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 32
sudo ./utilities/ovs-ofctl add-flow br0
  in_port=34,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 35
sudo ./utilities/ovs-ofctl add-flow br0
  in_port=33,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 17
```
Configurations: VM-VM KNI (Guest)

Host QEMU* command line VM1:

```
sudo $QEMU_DIR/x86_64-softmmu/gemu-system-x86_64 -c 0x30 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm -name "client 1" -nographic -vnc :1 $(cat /tmp/.ovs_ivshmem_gemu_cmdline_vm_1) -drive file=fat:rw:$QEMU_SHARE_DIR,snapshot=off -monitor unix:$VM1MONITOR,server,nowait &
```

Guest Configuration VM1

Initial guest setup as per ‘Configuration: KNI Guest Common’.

```
./build/kni_client -c 0x1 -n 4 -- -p ovskni0 -p ovskni2 &
ifconfig vEth_ovskni0 hw ether 00:4B:4E:49:30:00
ifconfig vEth_ovskni2 hw ether 00:4B:4E:49:30:02
ifconfig vEth_ovskni0 2.2.2.1/24 up
ifconfig vEth_ovskni2 4.4.4.1/24 up
sysctl -w net.ipv4.ip_forward=1
sysctl -w net.ipv4.conf.all_rp_filter=0
sysctl -w net.ipv4.conf.vEth_ovskni0_rp_filter=0
route add default gw 4.4.4.2 vEth_ovskni2
ifconfig -a
arp -s 4.4.4.2 DE:AD:BE:EF:CA:FE
```
Configurations: VM-VM KNI (Guest) (Contd.)

Host QEMU* command line VM2:
```
sudo $QEMU_DIR/x86_64-softmmu/qemu-system-x86_64 -c 0xC0 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm -name "client 2" -nographic -vnc :2 $(cat /tmp/.ovs_ivshmem_qemu_cmdline_vm_2) -drive file=fat:rw:$QEMU_SHARE_DIR,snapshot=off -monitor unix:$VM2MONITOR,server,nowait &
```

Guest Configuration VM2

Initial guest setup as per ‘Configuration: KNI Guest Common’.

```
./build/kni_client -c 0x1 -n 4 -- -p ovskni1 -p ovskni3 &
ifconfig vEth_ovskni1 hw ether 00:4B:4E:49:30:01
ifconfig vEth_ovskni3 hw ether 00:4B:4E:49:30:03
ifconfig vEth_ovskni3 4.4.4.2/24 up
ifconfig vEth_ovskni1 3.3.3.1/24 up
sysctl -w net.ipv4.ip_forward=1
sysctl -w net.ipv4.conf.all rp_filter=0
sysctl -w net.ipv4.conf.vEth_ovskni3 rp_filter=0
route add default gw 3.3.3.2 vEth_ovskni1
ifconfig -a
arp -s 3.3.3.2 DE:AD:BE:EF:CA:FE
```

Affinitize host cores, as described in ‘Configuration: Core Affinity’
VM Loopback US-Vhost

RESULTS
Intel® Xeon® E5-2680 Processor (DP)/Intel® C602 Chipset (Intel® BD82C602 PCH): IPv4 switching with 13-tuple lookup

**Intel® DPDK vSwitch Packet Switching: VM Loopback US-Vhost**

- **Packet size in bytes:**
  - 64: 7.43
  - 256: 4.53
  - 512: 2.35
  - 1024: 1.197
  - 1518: 0.813

Date: March 2014

Disclaimer: Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Source: Intel internal testing as of March, 2014. See Linux* Performance Tuning for configuration details.


Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.
Configurations: VM Loopback  US-Vhost (Host)

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section.

Initial setup, as per ‘Configuration: General Test Setup’, and ‘Configuration: US-Vhost Host Common’.

cd $OPENVSWITCH_DIR
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db
    ./vswitchd/vswitch.ovsschema
sudo ./ovsdb/ovsdb-server --remote=punix:/usr/local/var/run/openvswitch/db.sock --
    remote=db:Open_vSwitch,Open_vSwitch,manager_options &

    sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk
    sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy16 -- set Interface ovsphy16
        type=dpdkphy ofport_request=16 option:port=0
    sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy17 -- set Interface ovsphy17
        type=dpdkphy ofport_request=17 option:port=1
    sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost80 -- set Interface ovsvhost80
        type=dpdkvhost ofport_request=80
    sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost81 -- set Interface ovsvhost81
        type=dpdkvhost ofport_request=81

    sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --socket-mem
        2048,2048 --base-virtaddr=0x2aaa2aa0000 -- -p 0x03 -n 3 -h 2 --stats=0 --vswitchd=0 --
        -client_switching_core=1 --config="(0,0,2),(1,0,3)"
    sudo ./vswitchd.ovs-vswitchd -c 0x100 --proc-type=secondary &
Configurations: VM Loopback US-Vhost (Guest)

Copy Intel® DPDK to a temporary folder:
```bash
sudo rm -rf <QEMU_SHARE_DIR>
sudo mkdir -p <QEMU_SHARE_DIR>
mkdir <QEMU_SHARE_DIR>/DPDK
chmod 777 <QEMU_SHARE_DIR>
cp -a /path/to/DPDK/* <QEMU_SHARE_DIR>/DPDK
```

QEMU* command line
```bash
sudo $QEMU_DIR/x86_64-softmmu/qemu-system-x86_64 -c 0x30 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm --name "client 1" -nographic -vnc :1 -monitor unix:$VM1MONITOR,server,nowait -net none -no-reboot -mem-path /dev/hugepages -mem-prealloc -netdev typ e=tap,id=net1,script=no,downscript=no,ifname=ovsvhost80,vhost=on -device virtio-net-pci,netdev=net1 ,mac=00:00:00:00:00:01,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -netdev type=tap ,id=net2,script=no,downscript=no,ifname=ovsvhost81,vhost=on -device virtio-net-pci,netdev=net2,mac= 00:00:00:00:00:02,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -drive file=fat:rw:$QEMU_SHARE_DIR,snapshot=off
```
Configurations: VM Loopback US-Vhost (Guest) (Contd.)

```bash
mkdir -p /mnt/vhost_client
mount -o iocharset=utf8 /dev/sdb1 /mnt/vhost_client
mkdir -p /root/vhost_client
cp -a /mnt/vhost_client/* /root/vhost_client
cd /root/vhost_client/DPDK
export CC=gcc
export RTE_SDK=/root/vhost_client/DPDK
export RTE_TARGET=x86_64-ivshmem-linuxapp-gcc
make install T=x86_64-ivshmem-linuxapp-gcc

modprobe uio
insmod x86_64-ivshmem-linuxapp-gcc/kmod/igb_uio.ko
./tools/pci_unbind.py -b igb_uio 0000:00:03.0 0000:00:04.0
cd /root/vhost_client/DPDK/app/test-pmd
make clean
make
./testpmd -c 0x3 -n 4 --socket-mem 128 -- --burst=64 -i
```

At the “testpmd” prompt enter:
set fwd mac_retry
start

Affinitize cores, as described in ‘Configuration: Core Affinity’
VM-VM US-Vhost

RESULTS
Intel® Xeon® E5-2680 Processor (DP)/
Intel® C602 Chipset (Intel® BD82C602 PCH):
IPv4 switching with 13-tuple lookup

Intel® DPDK vSwitch Packet Switching:
VM-VM US-Vhost

Date: March 2014

Disclaimer: Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.
Source: Intel internal testing as of March, 2014. See Linux* Performance Tuning for configuration details.

For more information go to http://www.intel.com/performance

Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.
Configurations: VM-VM US-Vhost (Host)

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section.
Initial setup, as per ‘Configuration: General Test Setup’, and ‘Configuration: US-Vhost Host Common’.

```
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db
    ./vswitchd/vswitch.ovsschema
sudo ./ovsdb/ovsdb-server --remote=punix:/usr/local/var/run/openvswitch/db.sock --
    remote=db:Open_vSwitch,Open_vSwitch,manager_options &
sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy16 -- set Interface ovsphy16
    type=dpdk ofport_request=16
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy17 -- set Interface ovsphy17
    type=dpdk ofport_request=17
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost80 -- set Interface ovsvhost80
    type=dpdk ofport_request=80
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost81 -- set Interface ovsvhost81
    type=dpdk ofport_request=81
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost82 -- set Interface ovsvhost82
    type=dpdk ofport_request=82
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost83 -- set Interface ovsvhost83
    type=dpdk ofport_request=83
```
Configurations: VM-VM US-vHost (Host) (Cont.)

`sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --base-virtaddr=0x2aaa2aa0000 --socket-mem 2048,2048 -- -p 0x3 -n 2 -h 4 --stats=0 --vswitchd=0 --client_switching_core=1 --config="(0,0,2),(1,0,3)"

sudo ./vswitchd/ovs-vswitchd -c 0x100 --proc-type=secondary &

sudo ./utilities/ovs-ofctl add-flow br0
  in_port=16,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 80

sudo ./utilities/ovs-ofctl add-flow br0
  in_port=81,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 82

sudo ./utilities/ovs-ofctl add-flow br0
  in_port=84,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 17`
Configurations: VM-VM US-Vhost (Guest)

QEMU* command line VM1
sudo $QEMU_DIR/x86_64-softmmu/qemu-system-x86_64 -c 0x30 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm -name "client 1" -nographic -vnc :1 -monitor unix:$VM1MONITOR,server,nowait -net none -no-reboot -mem-path /dev/hugepages -mem-prealloc -netdev typ e=tap,id=net1,script=no,downscript=no,ifname=ovsvhost80,vhost=on -device virtio-net-pci,netdev=net1 ,mac=00:00:00:00:00:01,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -netdev type=tap ,id=net2,script=no,downscript=no,ifname=ovsvhost81,vhost=on -device virtio-net-pci,netdev=net2,mac= 00:00:00:00:00:02,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -drive file=fat:rw:$QEMU_SHARE_DIR, snapshot=off

QEMU* command line VM2
sudo $QEMU_DIR/x86_64-softmmu/qemu-system-x86_64 -c 0xC0 --proc-type secondary -n 4 -- -cpu host -boot c -hda $IMAGES/$IMAGE_NAME -snapshot -m 8192 -smp 2 --enable-kvm -name "client 2" -nographic -vnc :2 -monitor unix:$VM1MONITOR,server,nowait -net none -no-reboot -mem-path /dev/hugepages -mem-prealloc -netdev typ e=tap,id=net3,script=no,downscript=no,ifname=ovsvhost82,vhost=on -device virtio-net-pci,netdev=net3 ,mac=00:00:00:00:00:03,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -netdev type=tap ,id=net4,script=no,downscript=no,ifname=ovsvhost83,vhost=on -device virtio-net-pci,netdev=net4,mac= 00:00:00:00:00:04,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -drive file=fat:rw:$QEMU_SHARE_DIR, snapshot=off
Configurations: VM-VM US-Vhost (Guest) (Contd.)

Configure each client as described in ‘Configuration: VM Loopback US-Vhost (Guest)’.

Affinitize cores, as described in ‘Configuration: Core Affinity’
RESULTS

VM Loopback US-Vhost-VirtIO
Intel® Xeon® E5-2680 Processor (DP)/ Intel® C602 Chipset (Intel® BD82C602 PCH): IPv4 switching with 13-tuple lookup

Intel® DPDK vSwitch Packet Switching: VM Loopback US-Vhost-Virtio

<table>
<thead>
<tr>
<th>Packet size in bytes</th>
<th>Million Packets per Second</th>
</tr>
</thead>
<tbody>
<tr>
<td>64</td>
<td>0.931</td>
</tr>
<tr>
<td>256</td>
<td>0.917</td>
</tr>
<tr>
<td>512</td>
<td>0.915</td>
</tr>
<tr>
<td>1024</td>
<td>0.904</td>
</tr>
<tr>
<td>1518</td>
<td>0.813</td>
</tr>
</tbody>
</table>

Date: March 2014

Disclaimer: Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. Source: Intel internal testing as of March, 2014. See Linux* Performance Tuning for configuration details. For more information go to http://www.intel.com/performance

Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.
Configurations: VM Loopback

US-Vhost-Virtio (Host)

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section.

Initial setup, as per ‘Configuration: General Test Setup’, and ‘Configuration: US-Vhost Host Common’.

```bash
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db
   /vswitchd/vswitch.ovsschema
sudo ./ovsdb/ovsdb-server --remote=unix:/usr/local/var/run/openvswitch/db.sock --
   remote=db:Open_vSwitch,Open_vSwitch,manager_options &

sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy16 -- set Interface ovsphy16
type=dpdk ofport_request=16
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphy17 -- set Interface ovsphy17
type=dpdk ofport_request=17
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost80 -- set Interface ovsvhost80
type=dpdk ofport_request=80
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost81 -- set Interface ovsvhost81
type=dpdk ofport_request=81

sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --base-
virtaddr=0x2aaa2aa0000 socket_mem 2048,2048 -- -p 0x03 -h 2 -k 2 --stats=0 --
vswitchd=0 --client_switching_core=1 --config="(0,0,2),(1,0,3)"
```
Configurations: VM Loopback US-Vhost-VirtIO (Host) (Cont.)

```bash
sudo ./vswitchd/ovs-vswitchd -c 0x100 --proc-type=secondary &

sudo ./utilities/ovs-ofctl add-flow br0
    in_port=16,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 80

sudo ./utilities/ovs-ofctl add-flow br0
    in_port=81,dl_type=0x0800,nw_src=1.1.1.1,nw_dst=3.3.3.2,idle_timeout=0,action=output: 17
```
Configurations: VM Loopback US-Vhost-Virtio (Guest)

Host QEMU* command line:
sudo ./qemu/x86_64-softmmu/qemu-system-x86_64 -c 0x30 -n 4 --proc-type=secondary -- -cpu host -boot c -hda <PATH_TO_IMAGE>.qcow2 -m 512 -smp 2 --enable-kvm -name "Client 1" -nographic -vnc :1 -net none -no-reboot -mem-path /dev/hugepages -memprealloc -netdev type=tap,id=net1,script=no,downscript=no,ifname=ovsvhost80,vhost=on -device virtio-netpci,netdev=net1,mac=00:00:00:00:00:01,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -netdevtype=tap,id=net2,script=no,downscript=no,ifname=ovsvhost81,vhost=on -device virtio-netpci,netdev=net2,mac=00:00:00:00:00:02,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off

Guest Configuration:
ifconfig eth1 5.5.5.1/24 up
ifconfig eth2 90.90.90.1/24 up
sysctl -w net.ipv4.ip_forward=1
sysctl -w net.ipv4.conf.all.rp_filter=0
sysctl -w net.ipv4.conf.eth1.rp_filter=0
route add default gw 90.90.90.90 eth2
arp -s 90.90.90.90 DE:AD:BE:EF:CA:FE

Affinitize cores, as described in ‘Configuration: Core Affinity’
Configurations: VM Loopback US-Vhost-Virtio (Guest) (Contd.)

ifconfig eth1 5.5.5.1/24 up
ifconfig eth2 90.90.90.1/24 up
sysctl -w net.ipv4.ip_forward=1
sysctl -w net.ipv4.conf.all_rp_filter=0
sysctl -w net.ipv4.conf.eth1_rp_filter=0
route add default gw 90.90.90.90 eth2
arp -s 90.90.90.90 DE:AD:BE:EF:CA:FE

Affinitize cores, as described in ‘Configuration: Core Affinity’
VM-VM US-Vhost-VirtIO

RESULTS
Intel® Xeon® E5-2680 Processor (DP)/Intel® C602 Chipset (Intel® BD82C602 PCH): IPv4 switching with 13-tuple lookup

**Intel® DPDK vSwitch Packet Switching:** VM-VM US-Vhost-Virtio

![Graph showing packet switching performance](image)

**Date:** March 2014

**Disclaimer:** Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

**Source:** Intel internal testing as of March, 2014. See Linux* Performance Tuning for configuration details.

For more information go to [http://www.intel.com/performance](http://www.intel.com/performance)

Results have been measured by Intel based on software, benchmark or other data of third parties and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance. Intel does not control or audit the design or implementation of third party data referenced in this document. Intel encourages all of its customers to visit the websites of the referenced third parties or other sources to confirm whether the referenced data is accurate and reflects performance of systems available for purchase.
Configurations: VM-VM US-Vhost-Virtio (Host)

Platform, BIOS, Software, and Linux Environment configured as indicated in the ‘Configuration’ section.
Initial setup, as per ‘Configuration: General Test Setup’, and ‘Configuration: US-Vhost Host Common’.

```bash
sudo ./ovsdb/ovsdb-tool create /usr/local/etc/openvswitch/conf.db
   ./vswitchd/vswitch_ovsschema
sudo ./ovsdb/ovsdb-server --remote=punix:/usr/local/var/run/openvswitch/db.sock --
   remote=db:Open_vSwitch,Open_vSwitch,manager_options &

sudo ./utilities/ovs-vsctl --no-wait add-br br0 -- set Bridge br0 datapath_type=dpdk
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphyl6 -- set Interface ovsphyl6
   type=dpdk ofport_request=16
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsphyl7 -- set Interface ovsphyl7
   type=dpdk ofport_request=17
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost80 -- set Interface ovsvhost80
   type=dpdk ofport_request=80
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost81 -- set Interface ovsvhost81
   type=dpdk ofport_request=81
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost82 -- set Interface ovsvhost82
   type=dpdk ofport_request=82
sudo ./utilities/ovs-vsctl --no-wait add-port br0 ovsvhost83 -- set Interface ovsvhost83
   type=dpdk ofport_request=83
```
Configurations: VM-VM US-Vhost-Virtio (Host) (Contd.)

```
sudo ./datapath/dpdk/build/ovs_dpdk -c 0x0F -n 4 --proc-type primary --base-virtaddr=0x2aa2aa0000 socket-mem 2048,2048 -- -p 0x03 -h 4 -k 2 --stats=0 --vswitchd=0 --client_switching_core=1 --config="(0,0,2),(1,0,3)"

dssudo ./vswitchd/ovs-vswitchd -c 0x100 --proc-type=secondary &
```
Configurations: VM-VM US-Vhost-Virtio (Guest)

QEMU* command line VM1
sudo ./qemu/x86_64-softmmu/qemu-system-x86_64 -c 0x30 -n 4 --proc-type=secondary - -cpu host -boot c -hda <PATH_TO_IMAGE>.qcow2 -m 512 -smp 2 --enable-kvm -name "Client 1" -nographic -vnc :1 -net none -no-reboot -mem-path /dev/hugepages -memprealloc -netdev type=tap,id=net1,script=no,downscript=no,ifname=ovsvhost80,vhost=on -device virtio-netpci, netdev=net1,mac=00:00:00:00:00:01,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -netdev type=tap,id=net2,script=no,downscript=no,ifname=ovsvhost81,vhost=on -device virtio-netpci, netdev=net2,mac=00:00:00:00:00:02,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off

QEMU* command line VM2
sudo ./qemu/x86_64-softmmu/qemu-system-x86_64 -c 0xC0 -n 4 --proc-type=secondary - -cpu host -boot c -hda <PATH_TO_IMAGE>.qcow2 -m 512 -smp 2 --enable-kvm -name "Client 2" -nographic -vnc :2 -net none -no-reboot -mem-path /dev/hugepages -memprealloc -netdev type=tap,id=net3,script=no,downscript=no,ifname=ovsvhost82,vhost=on -device virtio-netpci, netdev=net3,mac=00:00:00:00:00:03,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off -netdev type=tap,id=net4,script=no,downscript=no,ifname=ovsvhost83,vhost=on -device virtio-netpci, netdev=net4,mac=00:00:00:00:00:04,csum=off,gso=off,guest_tso4=off,guest_tso6=off,guest_ecn=off
Configurations: VM-VM US-Vhost-Virtio (Guest) (Contd.)

Client 1 configuration
ifconfig eth1 up
ifconfig eth2 up
ifconfig eth1 5.5.5.1/24 up
ifconfig eth2 7.7.7.1/24 up
sysctl -w net.ipv4.ip_forward=1
sysctl -w
   net.ipv4.conf.all_rp_filter=0
sysctl -w
   net.ipv4.conf.eth1_rp_filter=0
route add default gw 7.7.7.2 eth2
arp -s 7.7.7.2 00:00:00:00:00:03

Client 2 configuration
ifconfig eth3 up
ifconfig eth4 up
ifconfig eth3 7.7.7.2/24 up
ifconfig eth4 6.6.6.1/24 up
sysctl -w net.ipv4.ip_forward=1
sysctl -w net.ipv4.conf.all_rp_filter=0
sysctl -w
   net.ipv4.conf.eth3_rp_filter=0
route add default gw 6.6.6.2 eth4
arp -s 6.6.6.2 DE:AD:BE:EF:FA:CE

Affinitize cores, as described in ‘Configuration: Core Affinity’
Summary

Observations

- Maximum 64B packet throughput of 10.99 Million Packets Per Second (Mpps).[3]

Notes

- The data is collected with best known configuration with NUMA aware port allocation. Any non-NUMA optimized configuration will result in performance degradation.
- Using different Core, NIC, and/or Port arrangements than those shown may impact performance.