Extending KVM Models Toward High-Performance NFV

Jun Nakajima, James Tsai, Mesut Ergin, Yang Zhang, and Wei Wang

14 October 2014
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.

Copyright © 2014 Intel Corporation.
Agenda

• The Challenge
• Architecture Proposals for NFV for KVM
• Current Status and Summary
NFV Vision from ETSI

Source: http://portal.etsi.org/nfv/nfv_white_paper2.pdf

Classical Network Appliance Approach

- Message Router
- CDN
- Session Border Controller
- WAN Acceleration
- DPI
- Firewall
- Carrier Grade NAT
- Tester/QoE monitor
- SGSN/GGSN
- PE Router
- BRAS
- Radio/Fixed Access Network Nodes

- Fragmented non-commodity hardware.
- Physical install per appliance per site.
- Hardware development large barrier to entry for new vendors, constraining innovation & competition.

Network Functions Virtualisation Approach

Figure 1: Vision for Network Functions Virtualisation
New/Different Requirements for NFV
Compared with Conventional Virtualization

• High performance across all packet sizes, including small packets (e.g. 64B)
• Real-time processing, including low latency and jitter
• RAS
• Security
• ...

Focus on Performance Topics Today
The Challenge

https://www.youtube.com/watch?v=qpfwDySweUA

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.
### Intel® DPDK Performance

* A snapshot of on different architectures

<table>
<thead>
<tr>
<th>Platform Features</th>
<th>Integrated Memory Controller</th>
<th>Data Direct I/O</th>
<th>4x10 GbE NICs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PCI-E Gen2</td>
<td>Integrated PCI-E Gen3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>AVX (integer, 128-bit)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Year</th>
<th>System Level L3 Performance (MPPS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009</td>
<td>42</td>
</tr>
<tr>
<td>2010</td>
<td>55</td>
</tr>
<tr>
<td>2011</td>
<td>93</td>
</tr>
<tr>
<td>2012</td>
<td>164.9</td>
</tr>
<tr>
<td>2013</td>
<td>255</td>
</tr>
</tbody>
</table>


https://www.youtube.com/watch?v=qpfwDySweUA

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.
Focus Areas for NFV Performance on KVM

Recall 67.2ns, 16.8ns, …

Fast and Efficient Inter-VM Communication

Generic: Network I/O, NUMA, NUMA-I/O, Caching, Affinity, …
Why Inter-VM Communication?

- More cores
- More middle boxes per socket, per server
- Service chaining on server
- Lower latency
- Inter-VM (i.e. intra-node) vs. Inter-node
- Higher Bandwidth
- Memory (or cache) vs. PCIe bus

Source (Figure 1.):
Inter-VM Communication on KVM

- Notifications for queue control
- Kick, Door Bell
- Virtual Switch
- Packet Transmission
- Copy, etc.
- Transitions
- User-Kernel
- Guest-Host

Switching path can be a big performance bottleneck

*Intel internal measurements
64B packets, virtio-net + vhost-net

0.712 Mpps*
0.717 Mpps*
Cost of Transitions/Isolation
Perspective of CPU Cycles

TSC Cycles (Haswell 3.2GHz), Round Trip*:
• User<->Kernel (System Call) in VM (on KVM)
  • E.g. getppid(): 1300 (≈ 400ns)
• Guest<->Host (Hyper Call)
  • E.g. Null Hypercall: 1500-1600 (≈ 500ns)

To reach Saturation Line Rate (10GbE):
• If system call/Hyper call is used for each 64B packet transmission, we would need:
  • > 6-7 Cores**
  • 40GbE:
    • > 24-28 Cores?

Practically, those are rather lower bounds because batching is limited and actual packet processing in hypercalls overturns gain of batching.

*Intel internal measurements
**:400/67.2 = 5.9, 500/67.2 = 7.4

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.
Agenda

• The Challenge
• Architecture Proposals for NFV for KVM
• Current Status and Summary
Solutions: Empower Guests in a Safe Way
Avoid hypervisor interventions

1. Move knowledge and control for inter-VM communication to VMs

2. Allow VMs to access other VMs to share or access memory in a safe way
   • Provide VMs with “Protected Memory View”
     • Mapping itself is provided by the hypervisor

3. Allow VMs to use low-latency notification mechanisms w/o VM exits or interrupts
   • E.g. MONITOR/MWAIT, Posted Interrupt
Example: vhost-net Functionality in Guests
vhost-user is already there

**Motivation:**

- Why does a kernel module need to know about data structures for PV drivers in guests?
- Because we trust kernel or kernel modules only.
- What if we trust specific (part of) guests…
- Vhost-net in guest can avoid hypercalls if it can directly access destination guests (virtqueue, etc.)
High-Level Architecture for Fast Inter-VM Communication (w/o VT-d, SR-IOV)

1. Data Transmission
   - Fast Path can work with virtio-net or independently

2. Notification
   - Protected Memory View
     - Shared memory for synchronization
   - Low-Latency Notification
     - Direct Access to Guests In Protected Memory View

VM1
- Kernel
- virtio-net
- Fast Path

VM2
- Kernel
- virtio-net
- Fast Path

KVM
- Linux Kernel
- Vhost-net API
High-Level Architecture for Fast Inter-VM Communication (with VT-d, SR-IOV)

- **VM0**: Middle Box (e.g. virtual switch)
- **VM1**: Shared memory for synchronization
- **VM2**: Shared memory for synchronization
- **VM1** and **VM2** have Fast Path connections to their respective Kernels
- **VM0** has a Fast Packet Transmission connection
- Fast Packet Transmission can be in user-level
- **KVM**: VT-d, SR-IOV
Introducing VM Function 0: EPTP* Switching

- VMFUNC instruction with EAX = 0
- Value in ECX selects an entry from the EPTP (Extended-Page-Table Pointer) list
- Available in Ring 0-3, executed in guest
  - No VM exit
  - Can be virtualized if not available

*:Extended-Page-Table Pointer
EPTP Switching and Trampoline Code

- VMFUNC executed outside Trampoline Code will cause EPT violation at next instruction
- Hypervisor needs to restore Default EPT to deliver virtual interrupts
More Details: Transmitting Packets

```
start_xmit(skb, dev) {
    ...
    send(packets);
}

send(packet) {
    ...
    VMFUNC #0, EPTP;
    Tx(packets);
    VMFUNC #0, 0
    ...
}

start_xmit(skb, dev) {
    ...
    send(packets);
}

send(packet) {
    ...
    VMFUNC #0, EPTP;
    Tx(packets);
    VMFUNC #0, 0
    ...
}

Tx(packet) {
    move_data();
    notify();
}
```

- EPT Perm.
- Modify queue descriptors
- Move Data by Tx()
- Modify queue descriptors
- XWR
- X-R
- X-R
- X-R
- XWR
- XWR
- X-R
- Page Boundary
- Trampoline Code
- Protected View
- Source VM
- Destination VM
- Default View
- Alternate View
Low-Latency Notification

Known methods

- Posted Interrupt
  - Deliver virtual interrupts on destination guests w/o VM exits.
  - Already supported by KVM
    - Still requires VM exit on source guest
- MONITOR/MWAIT (Energy-Efficient Polling) between guests
  - The feature is not advertised on KVM today
  - Use variables on shared memory between source and destination
- PAUSE Loop (Polling) between guests
  - Lowest latency, but not energy efficient

In practice, combine Interrupt and Polling (like NAPI)
Practices for Performance

General

Minimize impact of TLB misses, cache misses:

• Large pages (both guest, EPT, VT-d), NUMA, IO-NUMA, Data Direct I/O

• E.g. LIFO memory pool

• Zero-copy

• E.g. Add source buffers mapping to EPT of destination
  • If EPT PTEs were not valid, no INVEPT is required
Practice for Performance

EPTP Switching

Frequency of VMFUNC operation:

- Cost of VMFUNC is about 150 TSC cycles (Haswell, 3.2 GHz)*
  - Around 50ns, and sensitive to TLB, caches
  - Recall 67.2ns, 16.8ns, …

To reach Saturation Line Rate (10GbE):

- If VMFUNC is called for each 64B packet transmission, we
  - > 1-2 Cores (100ns for round-trip)
- 40GbE:
  - > 4-8 Cores?
- The cost of VMFUNC would be relatively small, and it would provide scalable performance

Constraint:

- getppid() in VM: 1300 (≈ 400ns)
- Null Hypercall: 1500-1600 (≈ 500ns)

Practically, those are rather lower bounds because batching is limited and actual packet processing overturns gain of batching.

*Intel internal measurements

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.
Security Consideration

• Trampoline Code is loaded by the guest, but the EPT permission (X-R) is set by KVM

• Should be signed together with the code in the Protected View in advance

• The set of pages (in Destination VM) accessed by code in Protected View need to be checked and added by KVM

• In a way, code in Protected View is an extension of the KVM/hypervisor running in controlled environment (still in VXM non-root mode)
Agenda

• The Challenge
• Architecture Proposals for NFV for KVM
• Current Status and Summary
Current Status

PoC

PoC in progress:

- Measured cost of VMFUNC, memory bandwidth
- Enabled and measured latency of MONITOR/MWAIT in guests
- Measuring path A
- Working on path B
Summary

Benefits of the Architecture:
• Contain knowledge and control for Inter-VM communication in guests
• Allow KVM to enable more optimization and customization for guests to handle high network loads efficiently
• More efficient and scalable than existing ones
• Work with direct I/O assignment as well

Next Step:
• Complete PoC and get more data
Backup
#VE: Virtualization Exception

- Can occur only in guest (vector 20)
- Some EPT violations can generate #VE instead of VM exits (controlled by hypervisor)
- Can virtualized if not available