High-Performance Virtualization for HPC Cloud on Xen

Jun Nakajima
Tianyu Lan
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 © 2016 Intel Corporation.
Agenda

• Intel® Xeon Phi™ processor
• HPC Cloud usage
• Challenges for Xen
• Achieving high performance
• Call for action
The world is going parallel – stick with sequential code and you will fall behind.

<table>
<thead>
<tr>
<th></th>
<th>Intel® Xeon® Processor E5-2600 v3 Product Family formerly codenamed Haswell</th>
<th>Intel® Xeon Phi™ x100 Product Family formerly codenamed Knights</th>
<th>Intel® Xeon® Processor E5-2600 v4 Product Family codenamed Broadwell</th>
<th>Intel® Xeon Phi™ x200 Product Family codenamed Knights Landing</th>
<th>Skylake</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cores</td>
<td>18</td>
<td>61</td>
<td>22</td>
<td>72</td>
<td>28</td>
</tr>
<tr>
<td>Threads/Core</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Vector Width</td>
<td>256-bit</td>
<td>512-bit</td>
<td>256-bit</td>
<td>512-bit (x2)</td>
<td>512-bit</td>
</tr>
<tr>
<td>Peak Memory Bandwidth</td>
<td>68 GB/s</td>
<td>352 GB/s</td>
<td>77 GB/s</td>
<td>&gt;500 GB/s</td>
<td>128 GB/s</td>
</tr>
</tbody>
</table>

The world is going parallel
Intel® Xeon Phi™ Processor

- Intel’s first bootable host processor specifically designed for HPC
- Binary compatible with Xeon Processor
- Integration of memory on package: Innovative memory architecture for high bandwidth and high capacity
- Integration of Omni-path Fabric on package
Intel® Xeon Phi™ Product Family

Available Today
Knights Corner
Intel® Xeon Phi™ x100 Product Family
- 22 nm process
- Coprocessor only
- >1 TF DP Peak
- Up to 61 Cores
- Up to 16GB GDDR5

Launched
Knights Landing
Intel® Xeon Phi™ x200 Product Family
- 14 nm process
- Host Processor & Coprocessor
- >3 TF DP Peak¹
- Up to 72 Cores
- Up to 16GB HBM
- Up to 384GB DDR4²
- ~460 GB/s STREAM
- Integrated Fabric²

Future
Knights Hill
3rd generation
- 10 nm process
- Integrated Fabric (2nd Generation)
- In Planning...

¹Results will vary. This simplified test is the result of the distillation of the more in-depth programming guide found here: https://software.intel.com/sites/default/files/article/3839671s-xeon-phi-right-for-me.pdf
²All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.
¹ Over 3 Teraflops of peak theoretical double-precision performance is preliminary and based on current expectations of cores, clock frequency and floating point operations per cycle. FLOPS = cores x clock frequency x floating point operations per second per cycle.
Hardware Overview

**Chip:**
Up To 36 tiles interconnected by Mesh

**Tile:**
2 Cores + 2 VPU/core + 1MB L2

**Core:**
4 hyper threads / core

**ISA:**
Binary Compatible with Intel Xeon processors + AVX 512 extension

**Memory:**
Up To 16GB on-package MCDRAM + up to 6 channels of DDR4-2400 (up to 384GB)

**IO:**
36 lanes PCIe Gen3 + 4 lanes DMI for chipset

**Node:**
1-socket only
# MCDRAM Memory modes

<table>
<thead>
<tr>
<th>Description</th>
<th>Cache Mode</th>
<th>Flat Mode</th>
<th>Hybrid Mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware automatically manages the MCDRAM as a “memory side cache” between CPU and ext DDR memory</td>
<td>64B cache lines direct-mapped 16GB MCDRAM</td>
<td>8GB/16GB MCDRAM</td>
<td>8 or 12GB MCDRAM</td>
</tr>
<tr>
<td>Manually manage how the app uses the integrated on-package memory and external DDR for peak perf</td>
<td>Physical Address Up to 384 GB DRAM</td>
<td></td>
<td>8 or 4 GB MCDRAM</td>
</tr>
<tr>
<td>Joins the benefits of both Cache and Flat modes by segmenting the integrated on-package memory</td>
<td></td>
<td></td>
<td>DRAM</td>
</tr>
</tbody>
</table>
MCDRAM(Flat)

- Platform with 2 NUMA nodes
- Memory allocated in DDR by default
  - Keep low bandwidth data out of MCDRAM
- Apps explicitly allocates important data in MCDRAM
Agenda

• Intel® Xeon Phi™ processor
• HPC Cloud usage
• Challenges for Xen
• Achieving high performance
• Call for action
HPC Cloud usage

- Single VM on one machine
- Expose most host CPUs to VM
  - More than 255 VCPUs in VM
- Expose MCDRAM to VM
- Pass through Omni-path Fabric to VM
Agenda

• Intel® Xeon Phi™ processor
• HPC Cloud usage
• Challenges for Xen
• Achieving high performance
• Call for action
Challenges for Xen

- Support >255 VCPUs
  - Virtual IOMMU support
- Scalability
  - Scalability issue in tasklet subsystem
Support >255 VCPUs

- HVM guest supports 128 VCPUs
- X2APIC mode is required for >255 VCPUs
- Linux disables X2APIC mode when no IR (interrupt remapping)
- No Virtual IOMMU support in Xen
- > 255 VCPUs => X2APIC => IR => Virtual IOMMU
- Enable DMA translation first
  - Linux IOMMU driver can’t work without DMA translation
Virtual IOMMU

Dom0

Qemu

Dummy Xen-VIOMMU

Hypervisor

Virtual IOMMU

VM

Linux Kernel

IOMMU driver

Hvmloader

Xenstore

Hypercall

ACPI DMAR
Virtual IOMMU (DMA Translation)

Dom0

Qemu

Virtual PCI device

Memory Region

Dummy Xen-VIOMMU

VM

Linux Kernel

IOMMU driver

Hypervisor

IOVA

IOVA -> GPA

Virtual IOMMU

IOVA -> Target GPA

Shadow IOVA -> HPA

Hardware

Memory

Physical IOMMU

Physical PCI Device

DMA

IOVA -> HPA
Virtual IOMMU (IR)

Dom0

Qemu

Virtual PCI device

Inject VIRQ

Hypervisor

Hardware

VM

Linux kernel

Device Driver

IRQ subsystem

IR table

VLAPIC

VIOAPIC/VMSI

Physical PCI Device

Virtual IOMMU

IR table

IRQ Remapping

VIRQ

IRQ

Inject VIRQ
Challenge for Xen

- Support >255 VCPUs
  - Virtual IOMMU support
- **Scalability**
  - Scalability issue in tasklet subsystem
Scalability issue in tasklet subsystem

- Tasklist work lists are percpu data structures
- A global spin lock “tasklet_lock” protects all these lists
- Tasklet_lock becomes hot point when running heavy workload in VM
  - Take average 180k tsc count to acquire global lock (IO VM exit: 150k tsc count)
- Change tasklet_lock to percpu lock
Agenda

• Intel® Xeon Phi™ processor
• HPC Cloud usage
• Challenges for Xen
• Achieving high performance
• Call for action
Achieving high performance

- Expose key compute resources to VM:
  - CPU topology
  - MCDRAM
- Reduce timer interrupts
VM CPU topology

- HPC software assigns workload according to CPU topology
- Balance workload among physical cores
Expose MCDRAM to VM

- Create vNUMA nodes as host’s NUMA topology
- Keep vNUMA of MCDRAM with far distance to vNUMA of CPU
Reduce timer interrupts

• Local APIC timer interrupt causes frequent VM exit (26000 exits/s) during running benchmark
• Reduce timer interrupt via setting timer_slop to 10ms
• Side affect: Low timer’s resolution

![Benchmark Diagram]

- Stream: Original VM (87), Tasklet fixed VM (85), Timer slop VM (98)
- Dgemm: Host (63), Original VM (50), Tasklet fixed VM (50), Timer slop VM (97)
- Sgemm: Host (99), Original VM (86), Tasklet fixed VM (97), Timer slop VM (98)
Reduce timer interrupts (Next to do)

Hypervisor:
• No need scheduler for single VM

Guest:
• Make Guest Linux tickless
Agenda

• Intel® Xeon Phi™ processor
• HPC Cloud usage
• Challenges for Xen
• Achieving high performance
• Call for action
Call for action

• We were able to achieve high-performance HPC on Xen
• Changes required in Xen
  • Increase vcpu numbers
    › 128 => 255 vcpus
    › Virtual IOMMU