Intel Virtualization Technology Roadmap and VT-d Support in Xen

Jun Nakajima

Intel Open Source Technology Center

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.
-

Throughout this presentation:
VT-x *refers to Intel® VT for IA-32 and Intel® 64*
VT-i *refers to the Intel® VT for IA-64, and*
VT-d *refers to Intel® VT for Directed I/O*

Intel® VT Roadmap: Overview

Vector 3:
I/O Focus

PCI-SIG

Standards for I/O-device sharing:
- Natively sharable I/O devices
- Endpoint DMA-translation caching

Vector 2:
Platform Focus

VT-d

Infrastructure for I/O-device virtualization:
- DMA protection and remapping
- Interrupt filtering and remapping

Vector 1:
Processor Focus

VT-x

VT-i

Establish foundation for virtualization in the Intel® 64 and Itanium® architectures…

… followed by on going evolution of support:
- Microarchitectural (e.g., lower VM entry/exit costs)
- Architectural (e.g., extended page tables – EPT)

VMM
Software
Evolution

Software-only VMMs
- Binary translation
- Paravirtualization
- Device Emulation

Simpler and more **Secure** VMMs through foundation of virtualizable ISAs

Improved CPU and I/O virtualization **Performance** and **Functionality** as VMMs exploit infrastructure provided by VT-x, VT-i, VT-d

Past
No Hardware
Support

Today

VMM software evolution over time with hardware support

New Feature Highlights

- APIC TPR Virtualization
  - Significantly reduce VM exits caused by access to local APIC TPR (not CR8)
    - Submitted a patch (last month, not in yet)
- Virtual-processor Identifiers (VPIDs)
  - Supports retention of TLB entries across VM switches
- Extended page tables (EPT)
- NMI-window Exiting
  - Enables timely delivery of NMIs to guest OS

New Feature Highlights (cont.)

- Preemption Timer
  - Allows VMM to bound guest-OS execution time
- Descriptor-table Exiting
  - Enables VMM to protect IDT, GDT, etc. from attack in guest OS
- Interrupt remapping (VT-d2)

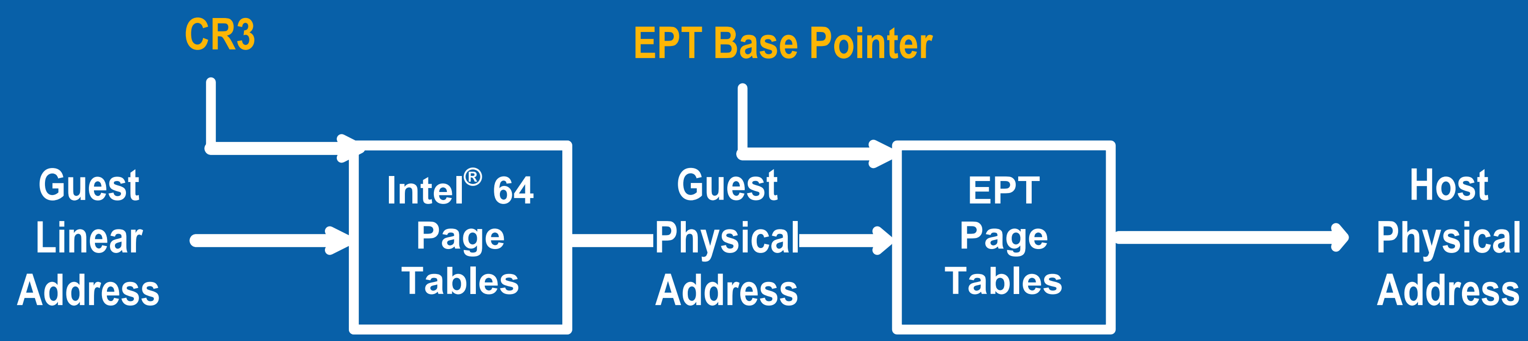
VPIDs: General Idea

- TLBs cache for multiple address spaces
- Address spaces distinguished by VPIDs
  - Host software runs with VPID zero
  - Each virtual CPU has its own non-zero VPID
- CPU uses VPIDs to prevent TLB sharing

VPIDs: Details

- New VM-execution controls:
  - Use VPID (single-bit control)
  - VPID value
- If use VPID is set:
  - Guest's VPID used while guest is executing
  - No TLB flushes on entry to or exit from guest
- If use VPID is clear:
  - Guest execution uses VPID zero
  - TLB flushes on entry and exit
- New instruction for VMM to flush per VPID

EPT: Overview



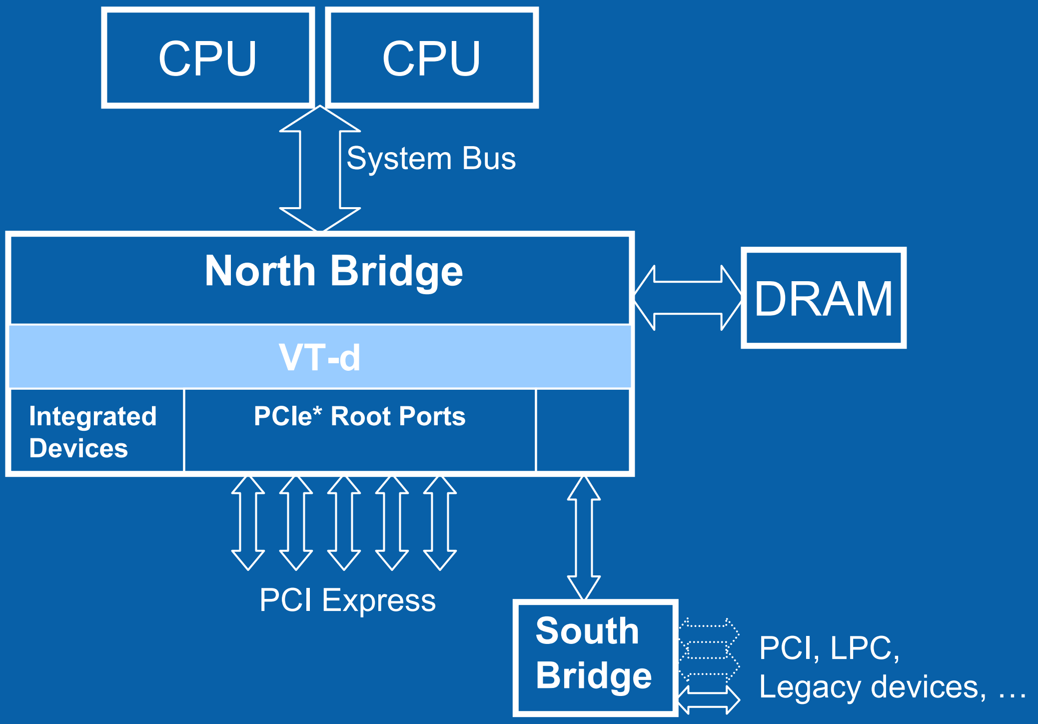
- Intel® 64 page tables
  - Map guest-linear to guest-physical (translated again)
  - Can be read and written by guest
- New EPT page tables under VMM control
  - Map guest-physical to host-physical (accesses memory)
  - Referenced by new EPT base pointer
- No VM exits due to page faults, INVLPG, or CR3 accesses

EPT Page Tables

- Page-table details similar to Intel® 64:
  - Each table has 512 8-byte entries (4KB)
  - 4 levels of page tables
  - Permission bits for read, write, execute
- Disallowed accesses
  - Called EPT violations
  - Cause VM exits

VT-d Overview

- VT-d provides infrastructure for I/O virtualization
  - Defines architecture for DMA and interrupt remapping
  - Common architecture across IA platforms
  - Will be supported broadly across Intel® chipsets



VT-d Applied to Pass-through Model

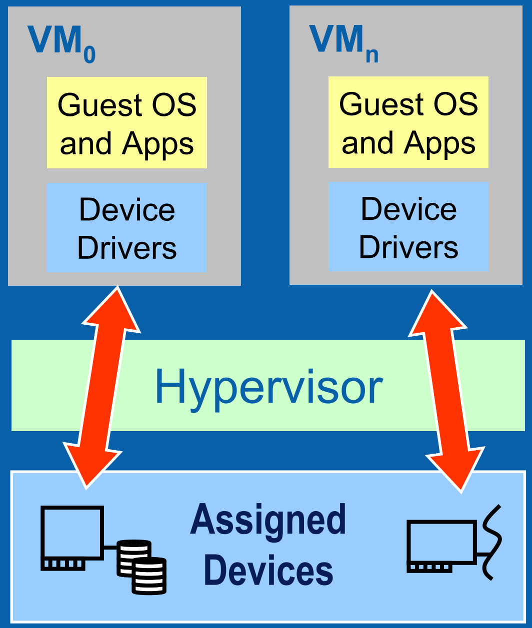
Direct Device Assignment to Guest OS

- Guest OS directly programs physical device
- For legacy guests, hypervisor sets up guest- to host-physical DMA mapping
- For remapping aware guests, hypervisor involved in map/unmap of DMA buffers

PCI-SIG I/O Virtualization Working Group

- Activity towards standardizing natively sharable I/O devices
- IOV devices provide virtual interfaces, each independently assignable to VMs

Pass-through Model



Pro: Highest Performance
Pro: Smaller Hypervisor
Pro: Device-assisted sharing
Con: VM Migration Limits

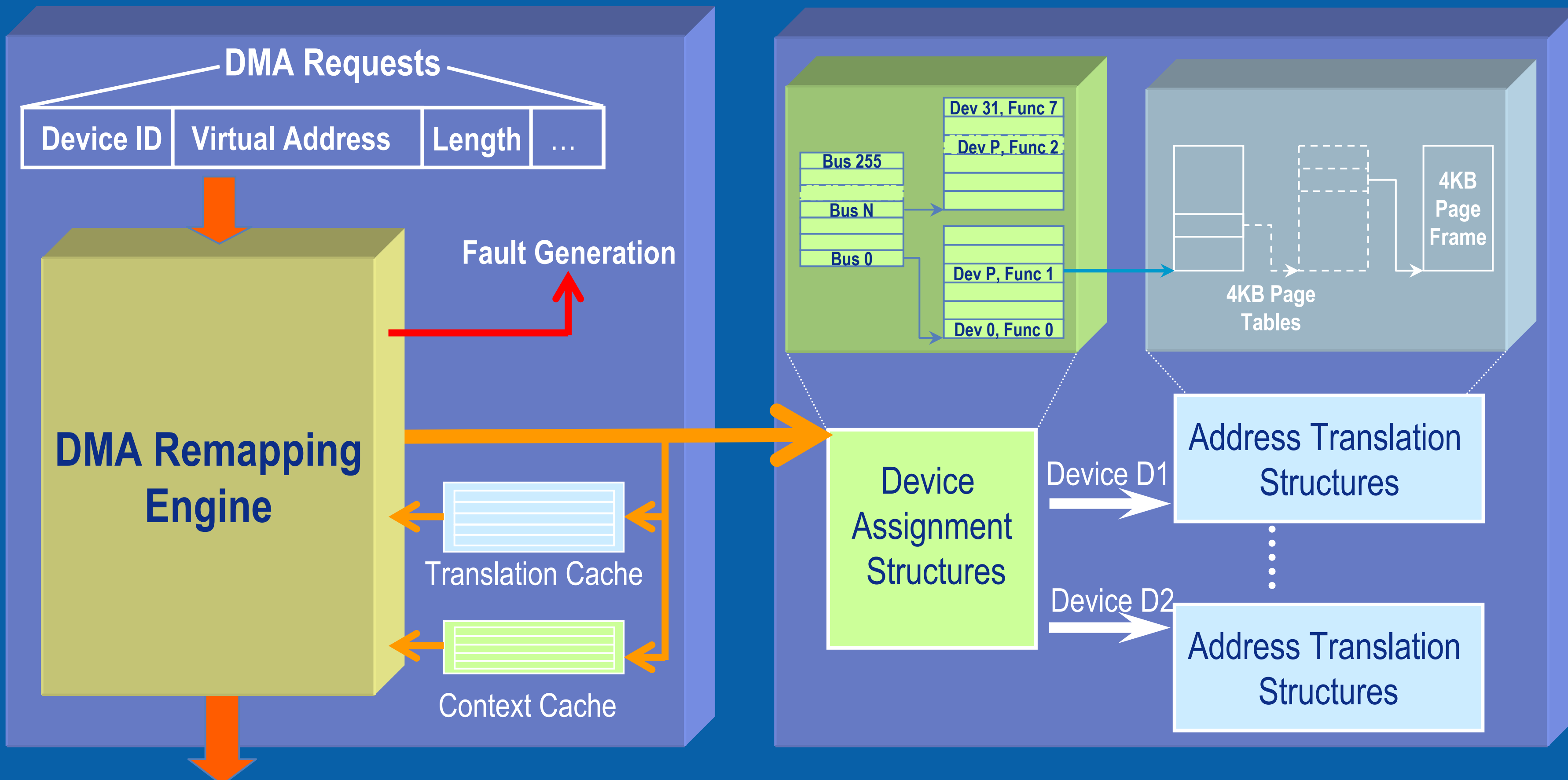
DMA Remapping: Features

- Translates DMA requests from all devices
  - DMA requests specify DMA Virtual Address
  - Hardware translates to Host Physical Address

- Flexible DMA virtual address space management
  - DMA address space per device or sharable across devices
  - Page granular memory management

- Other Features
  - H/W caching of frequently used remapping structures
  - Support for PCIe* Address Translation Services (ATS)
  - Improved RAS by reporting DMA faults to software

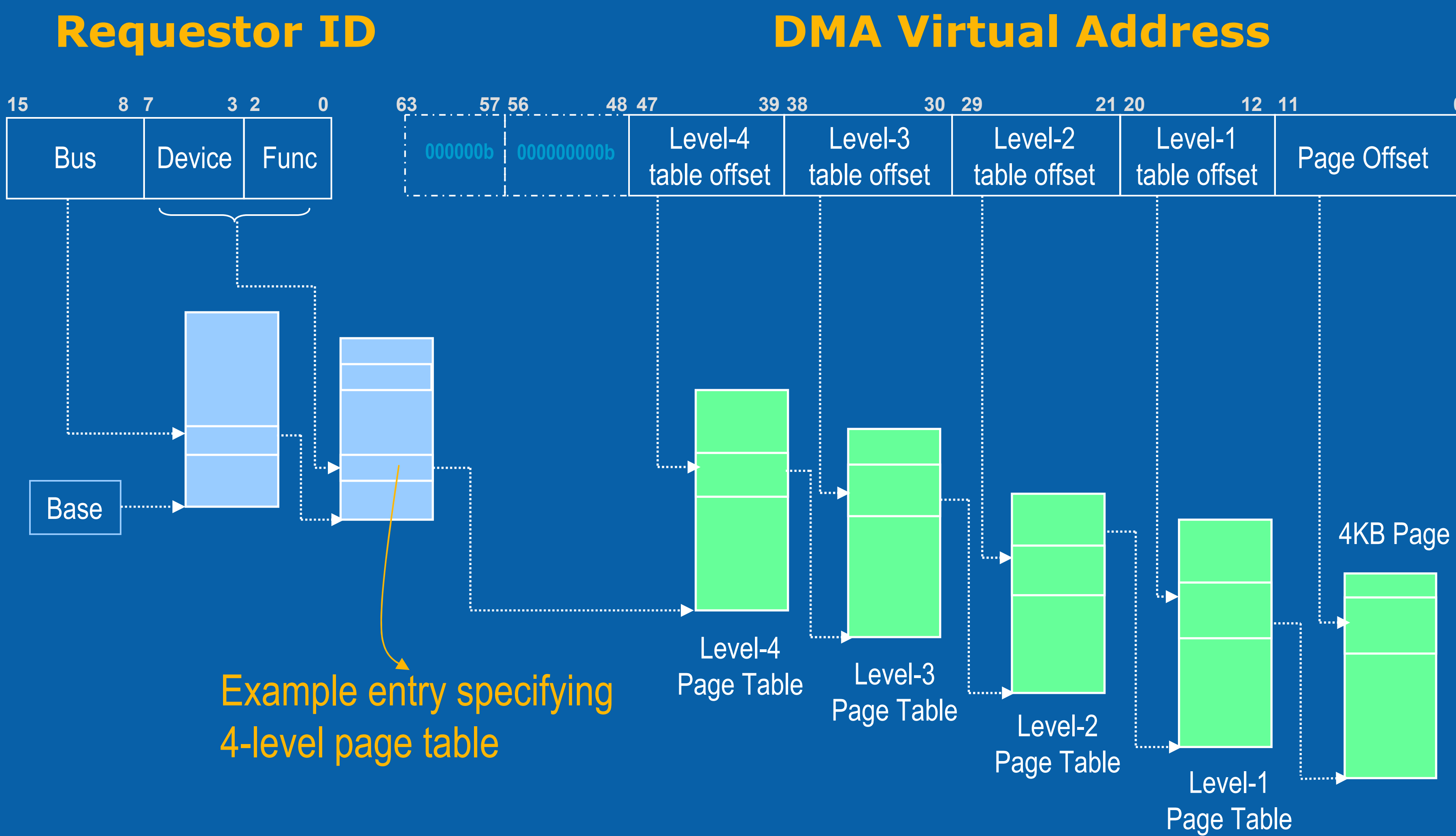
DMA Remapping: Hardware Overview



Memory Access with Host Physical Address

Memory-resident Partitioning & Translation Structures

DMA Remapping: Page Walk



Interrupt Virtualization

- Drivers for direct assigned devices run within VM
  - Driver only aware of virtual CPU of the VM
  - Device interrupts needs to be delivered to virtual CPU
  - VT-x provides support for virtual CPU interrupt delivery

- Support lacking to isolate & route device interrupts
  - Any direct assigned MSI capable device can generate any physical interrupt (no interrupt isolation)
  - No support to drain in-flight interrupts destined to a CPU
  - No easy way to re-direct device interrupts (require IPIs)

Interrupt remapping enables interrupt isolation and routing

Interrupt Remapping

- Interrupt request specify request & originator IDs
  - Remap hardware transforms request to physical interrupt
- Interrupt remapping hardware
  - Enforces isolation through use of originator ID
  - Generated interrupts with attributes in remap structure
  - Caches frequently used remap structures
  - S/W may modify remap for efficient interrupt re-direction
- Applicable to all interrupt sources
  - Legacy interrupts delivered through I/O APICs
  - Message signaled interrupts (MSI, MSI-X)
  - Works with existing device hardware

VT-d Support in Xen

- Device assignment by hypercalls
  - Device assignment
    - Give the ownership of the device
  - I/O port access
    - Unblock or remapping
  - IRQ mapping
    - Remap interrupts
  - MMIO handling
    - Set up translation in the shadow page table so that the guest can directly access the device memory
- PCI config space virtualization
  - BAR virtualization
- VT-d table for the device assigned
  - Detect VT-d via ACPI tables
  - Build (static) page tables for the device (BDF) using the P2M routines

Current Status

- Sanity Checks
  - Assigned PCIe E1000 add-on card to 32-bit FC5 on 64-bit Xen.
  - "scp" test shows near-native performance on the test machine (e.g 200+Mbps).
- Submitted the patches to xen-devl mailing list this month
- Testing on other guests

Leap ahead™