LINUXCON JAPAN

Making Nested Virtualization Real by Using Hardware Virtualization Features

May 28, 2013

Jun Nakajima
Intel Corporation

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.

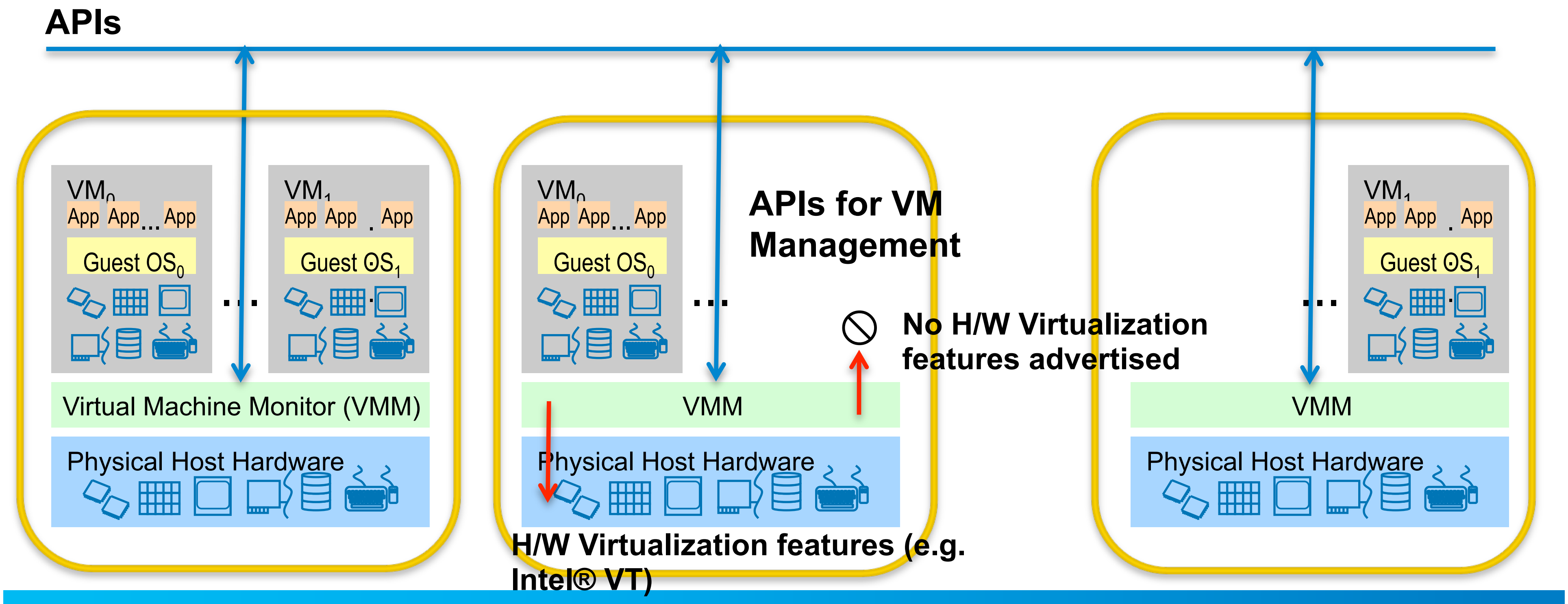
*Other names and brands may be claimed as the property of others.

Agenda

- Why does “Nested Virtualization” matter for cloud?
  - What is it?
  - How does it enhance the cloud?
- How is Nested Virtualization implemented?
  - What are the challenges?
  - Which hardware virtualization features are helpful?
- Current status
  - Performance and functionality
  - Summary

How Virtualization is Used in the Cloud?

- **Compute Nodes:**
  - Cloud software (e.g. OpenStack) uses API to manage VMs
  - VMMs (e.g. KVM, Xen, etc.) use H/W Virtualization features to run guests
  - H/W Virtualization features are not available for guests



APIs for VM Management

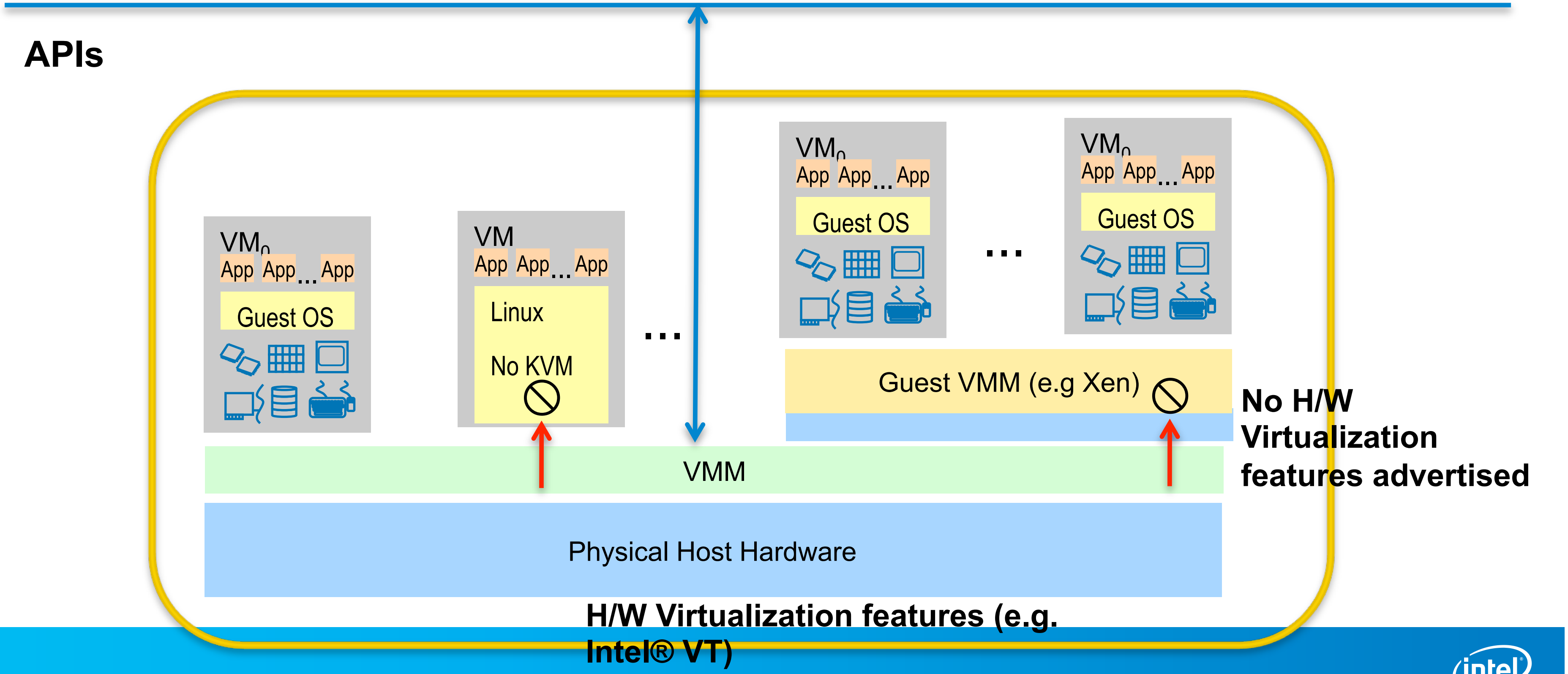
No H/W Virtualization features advertised

H/W Virtualization features (e.g. Intel® VT)

Compute Node

Lack of H/W Virtualization Features in Cloud Applications Means:

- **VMs:**
  - No KVM on Linux, No Hyper-V functionality on Windows
  - No HVM (Hardware-based VM) on Xen – e.g. No Windows support
  - Need to use software emulation – Very slow

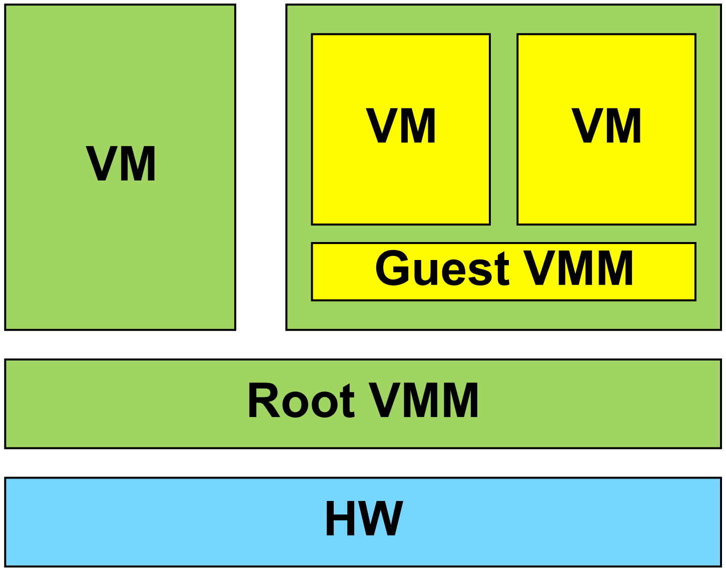


No H/W Virtualization features advertised

H/W Virtualization features (e.g. Intel® VT)

What’s “Nested Virtualization”?

- **Software feature in (Root) VMM that allows Guest VMM to use H/W virtualization features**
  - “Virtual” H/W virtualization features (nested)
  - May use H/W virtualization features



- **H/W virtualization features (e.g. Intel® VT)**
  - VT-x (CPU virtualization)
    - VXM instructions, VMCS (Virtual Machine Control Structure), EPT (Extended Page Table), etc.
  - VT-d (Direct I/O)
  - VT-c (Connectivity, especially SR-IOV of NIC)

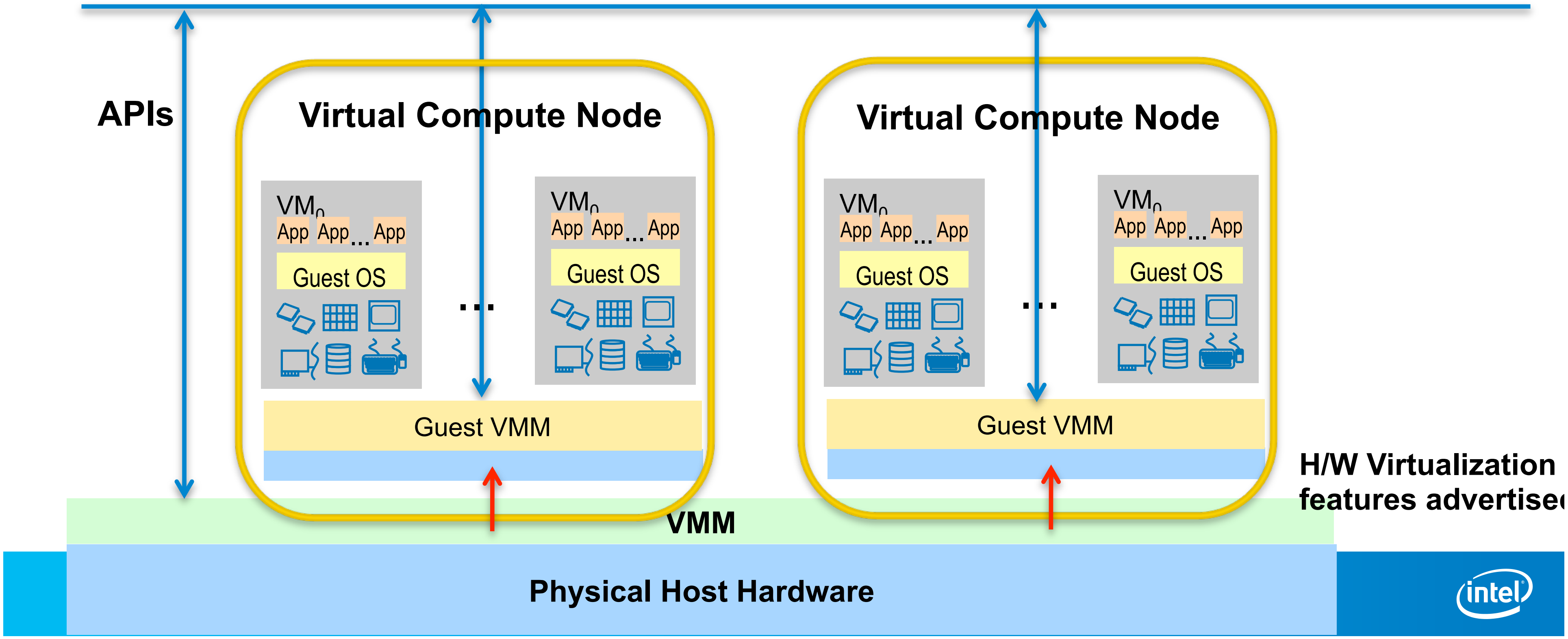
Motivations (1) – Enhance Hosting Capabilities/Features of the Cloud

- **Data centers using H/W virtualization products**
  - Cannot be hosted in clouds without nested virtualization
- **Operating systems with built-in H/W virtualization support**
  - Lose features (or fail back to software solutions)
    - XP mode for VDI
    - Hyper-V
- **System Emulators with H/W virtualization**
  - Run very slow in cloud (or fall back to software emulation)
    - Android Emulator on Linux (KVM) and Windows (HAXM*)

*: Intel® Hardware Accelerated Execution Manager
http://software.intel.com/en-us/articles/intel-hardware-accelerated-execution-manager

Motivations (2) – Cloud Virtualization

- **Hosting clouds with fewer physical severs**
  - More cores, dynamic resource utilization using Virtual Compute Nodes
- **Cloud Development**
  - Increase productivity, lower cost
- **Large-scale testing of cloud**
  - Improve security, quality



Agenda

- Why does “Nested Virtualization” matter for cloud?
  - What is it?
  - How does it enhance the cloud?
- How is Nested Virtualization implemented?
  - What are the challenges?
  - Which hardware virtualization features are helpful?
- Current status
  - Performance and functionality
  - Summary

Challenges of Nested Virtualization

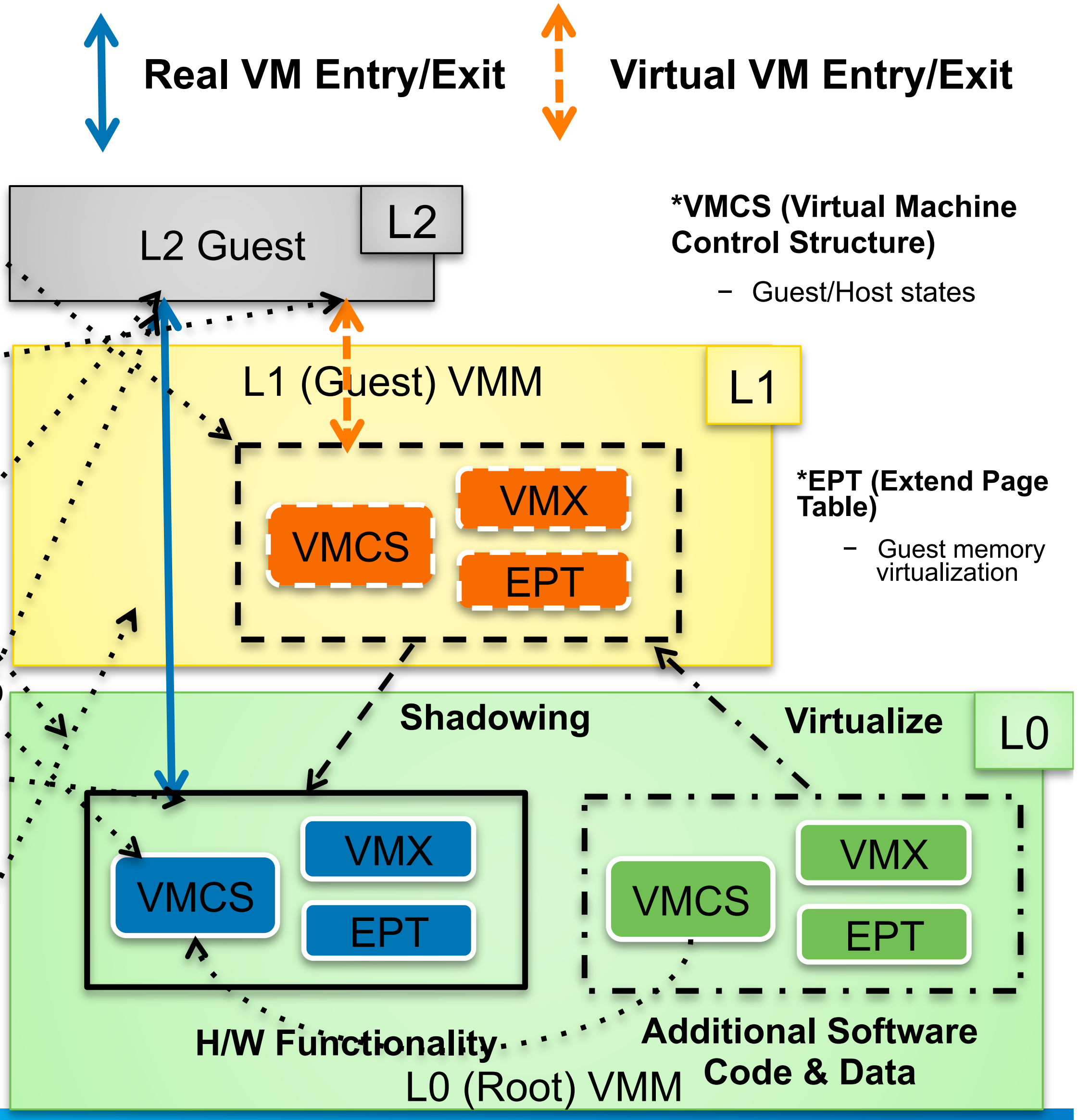
- **Extra Overheads – Potentially lower performance**
  - Higher VM Exit rates (Next slides)
  - Software overhead of virtualizing “H/W virtualization features”
- **Complexity of Root VMM Software**
  - More surface areas for security attacks
    - Sometimes exposes existing bugs with (Guest) VMMs
  - Requires more QA because of various combinations

Example of Extra Overheads – VM Entry/Exit

1. L1 creates VT-x structures for L2
2. L1 enters VM (Virtual VM Enter)

   VMLAUNCH, VMRESUME
3. Trapped by L0 (VM Exit)
4. L0 sets up real VMCS
5. L0 enters L2 VM (Real VM Enter)

   VMLAUNCH, VMRESUME
6. At some point L2 causes VM exit to L0 (Real VM Exit)
7. L0 handles VM Exit itself and resumes L2 (Real VM Enter) or injects VM Exit to L1 (Virtual VM Exit)
8. Repeat from 2.



*VMCS (Virtual Machine Control Structure)

- Guest/Host states

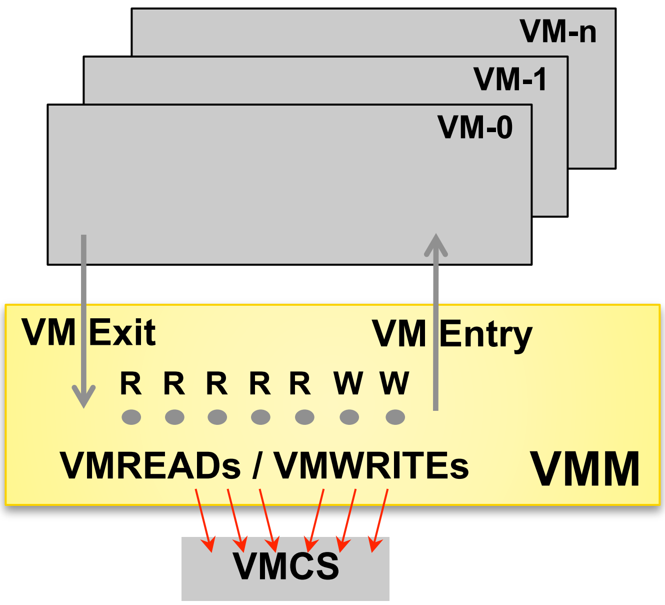
*EPT (Extend Page Table)

- Guest memory virtualization

Reducing Extra Overheads

Build Foil

Standard Virtualization

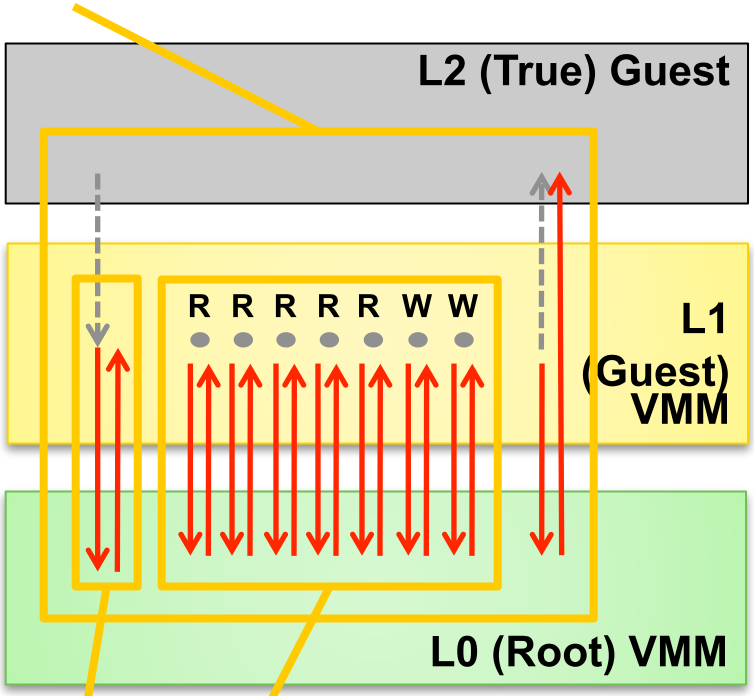


VMCS (Virtual Machine Control Structure)

- Holds guest and host CPU register state
- Increasingly optimized with each VT implementation
- The key to reducing VT latencies over time

Opportunity #2: Reduce “virtual” VM exits entirely (e.g., via EPT, APIC Virtualization)

Nested Virtualization



Opportunity #3: Eliminate VM exits on guest VMCS Accesses

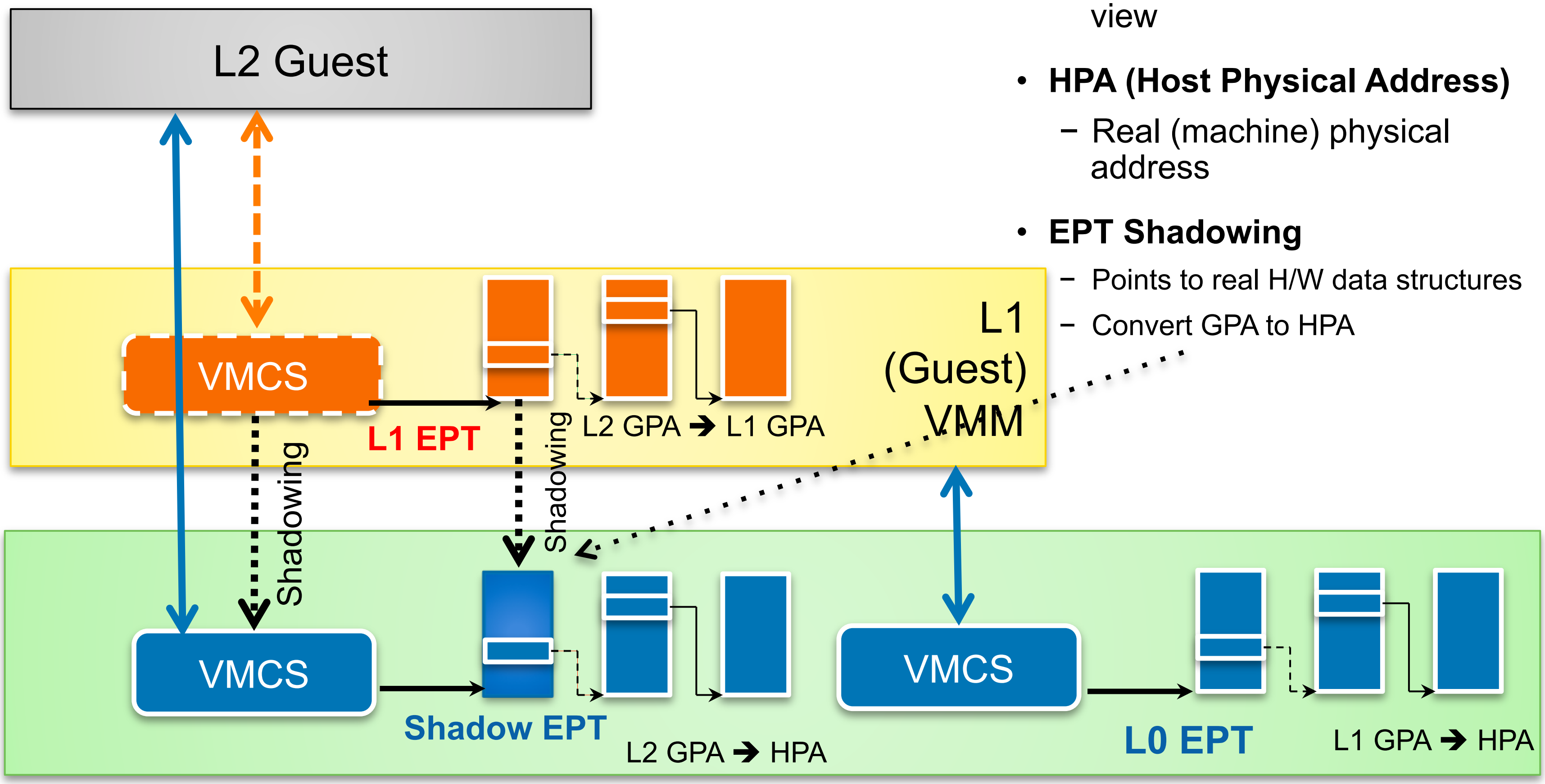
Opportunity #1: Reduce transition latencies

*KVM/Xen : 8+ VMREADs, 3+ VMWRITEs per VM Exit (Approximately, depends on the Exit type and version)

Improving Performance of Nested Virtualization (Recap)

- **Opportunity #1: Reduce Transition Latencies**
  - Reduce unique overheads of virtualization. Intel is fanatically committed.
  - Optimize software code
- **Opportunity #2: Reduce "Virtual" VM Exits Entirely**
  - EPT (Implemented as Virtual EPT)
  - APIC Virtualization
    - Eliminate or reduce VM exits with access to local APIC
    - Guest VMMs can access local APICs more frequently to virtualize timers, I/O devices
  - VT-d, SR-IOV
    - Reduce overhead of I/O virtualization
    - Guest VMMs can access I/O devices more frequently
- **Opportunity #3: Eliminate VM Exits on guest VMCS Accesses**
  - VMCS Shadowing

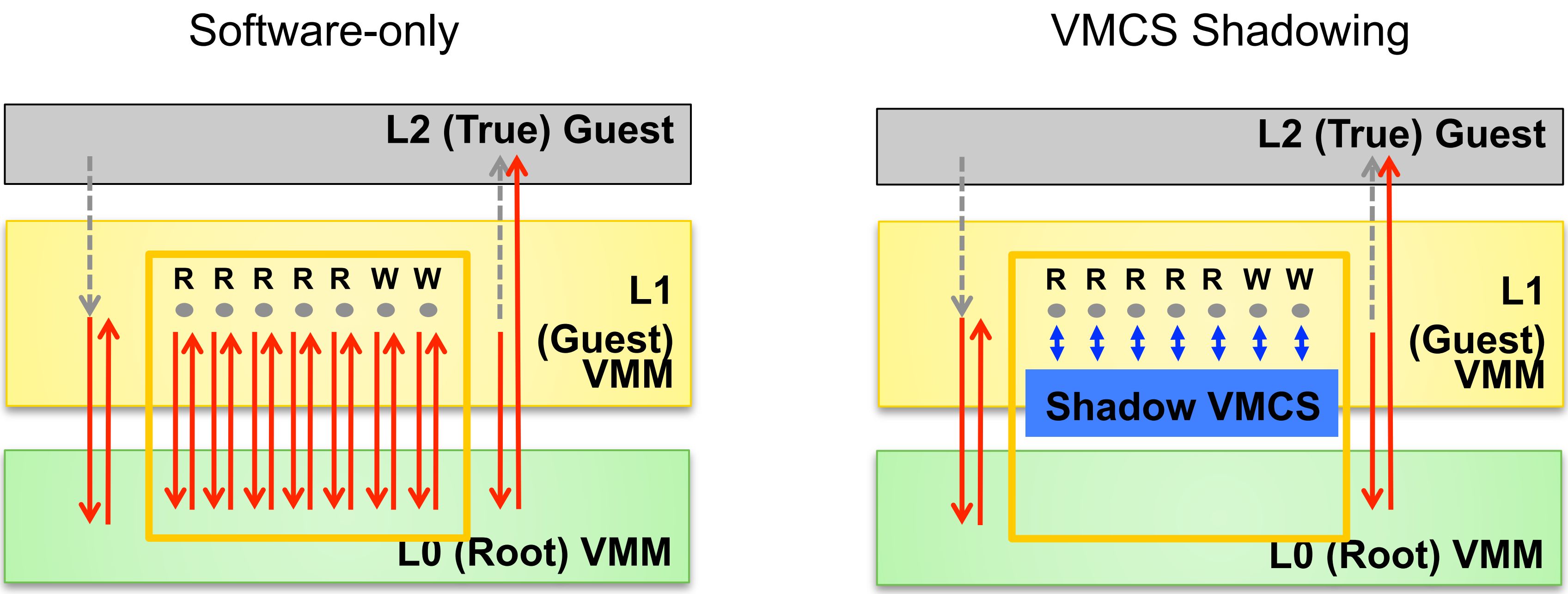
Virtual EPT



- **GPA (Guest Physical Address)**
  - Physical address in guest's view
- **HPA (Host Physical Address)**
  - Real (machine) physical address
- **EPT Shadowing**
  - Points to real H/W data structures
  - Convert GPA to HPA

Switch to Shadow EPT @ VM Entry to L2

New H/W Feature: VMCS Shadowing



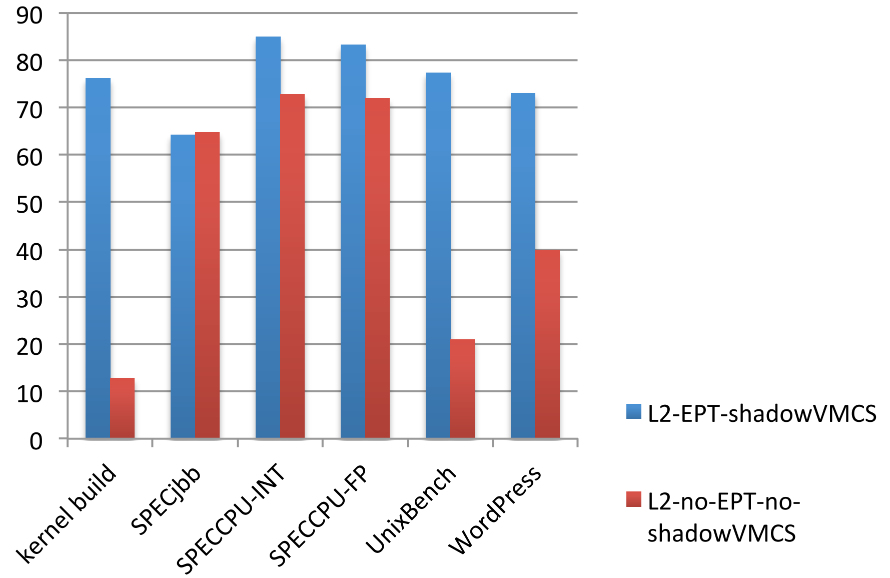
- **VMREAD-Bitmap and VMWRITE-Bitmap**
  - VM Exit if Bit *n* in VMREAD/VMWRITE bitmap is 1, where *n* is value of bits 14:0 of register source/destination operand
- **Direct Guest VMM VMREAD/VMWRITE to a Shadow VMCS**
  - Accesses to Shadow VMCS done by hardware
  - Eliminates majority of nesting-induced VM exits
  - Improves performance of software stacks that support nesting

Agenda

- Why does “Nested Virtualization” matter for cloud?
  - What is it?
  - How does it enhance the cloud?
- How is Nested Virtualization implemented?
  - What are the challenges?
  - Which hardware virtualization features are helpful?
- Current status
  - Performance and functionality
  - Summary

Performance Trending

- **With only virtual EPT and VMCS Shadowing, performance of L2 is around 80% of L1***
- **APIC Virtualization could provide approx. 3% additional improvement in Kernel Build and SPECCPU cases (not shown in the chart)***
- **Expect more gain from:**
  - VT-d, SR-IOV
- **Looking at issues with SPECjbb**



L2 (Linux) Performance relative to L1 (KVM on KVM)

WordPress:
L2 Guest OS: RHEL6.4 (with 4vCPUs and 4GB memory)
Web Server: Apache (httpd-2.2.15-26.el6.x86_64.rpm)
Database: MySQL (mysql-server-5.1.66-2.el6_3.x86_64.rpm)
Web App: WordPress v3.5.1
JMeter (a client on another machine): JMeter v2.9

*Estimated Results Benchmark Disclaimer

Results have been estimated based on internal Intel analysis and are provided for informational purposes only. Any difference in system hardware or software design or configuration may affect actual performance.

Current Status

- **Crucial features for Nested Virtualization in KVM and Xen**
  - Virtual EPT – KVM (WIP, v3 submitted), Xen (upstream)
  - VMCS Shadowing – KVM (upstream), Xen (upstream)
  - APIC Virtualization – KVM (upstream), Xen (upstream)
  - VT-d, SR-IOV – KVM (in distributions), Xen (in distributions)
- **Our Test Cases (KVM and Xen as L0)**
  - L1
    - KVM (L0 KVM and Xen)
    - Xen (on L0 Xen, issues on L0 KVM)
    - VMware Player 5.0 on Windows 7 (pass on L0 Xen, issues on L0 KVM)
    - VirutalBox 4.2 on Windows 7 (on L0 Xen) – Issues
  - L2
    - 32/64-bit Linux
    - 32/64-bit Windows

Summary

- **Nested Virtualization**
  - Extends hosting capabilities/features of cloud
  - Provides a means to virtualize cloud
  - Becoming realistic solutions with new H/W features and software support
- **Performance is getting closer to L1**
  - With only virtual EPT and VMCS Shadowing, performance of L2 is getting around 80% of L1*
  - More gains are expected with other H/W virtualization features
- **Functionality**
  - KVM on KVM, KVM/Xen on Xen
  - VMware on Xen, VMware on KVM (WIP)

Nested Virtualization Is Becoming Real

*Estimated Results Benchmark Disclaimer
Results have been estimated based on internal Intel analysis and are provided for informational purposes only.
Any difference in system hardware or software design or configuration may affect actual performance.