Nested Virtualization Update From Intel

Xiantao Zhang, Eddie Dong

Intel Corporation

August 27-28, 2012
San Diego, CA, USA

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.

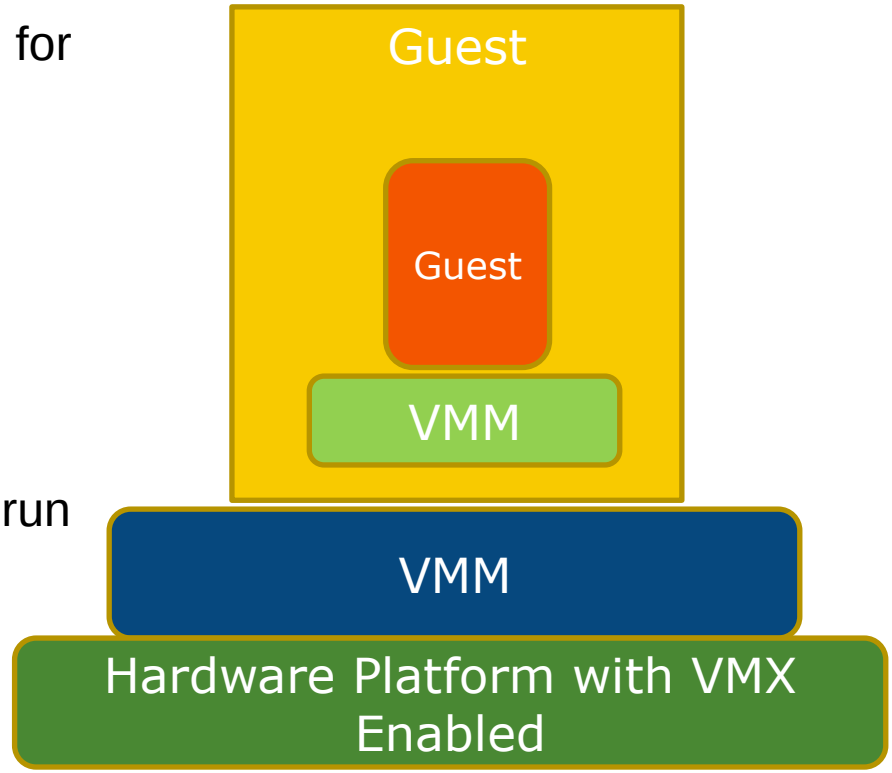
*Other names and brands may be claimed as the property of others.

Agenda

- Motivation and Goals
- History
  - Nested VMX Architecture
  - Previous status
- Latest status and new features
  - Stability Enhancement
  - Virtual EPT
  - Virtual VT-d
- Preliminary Performance
- Call to Action

Motivation and Goals

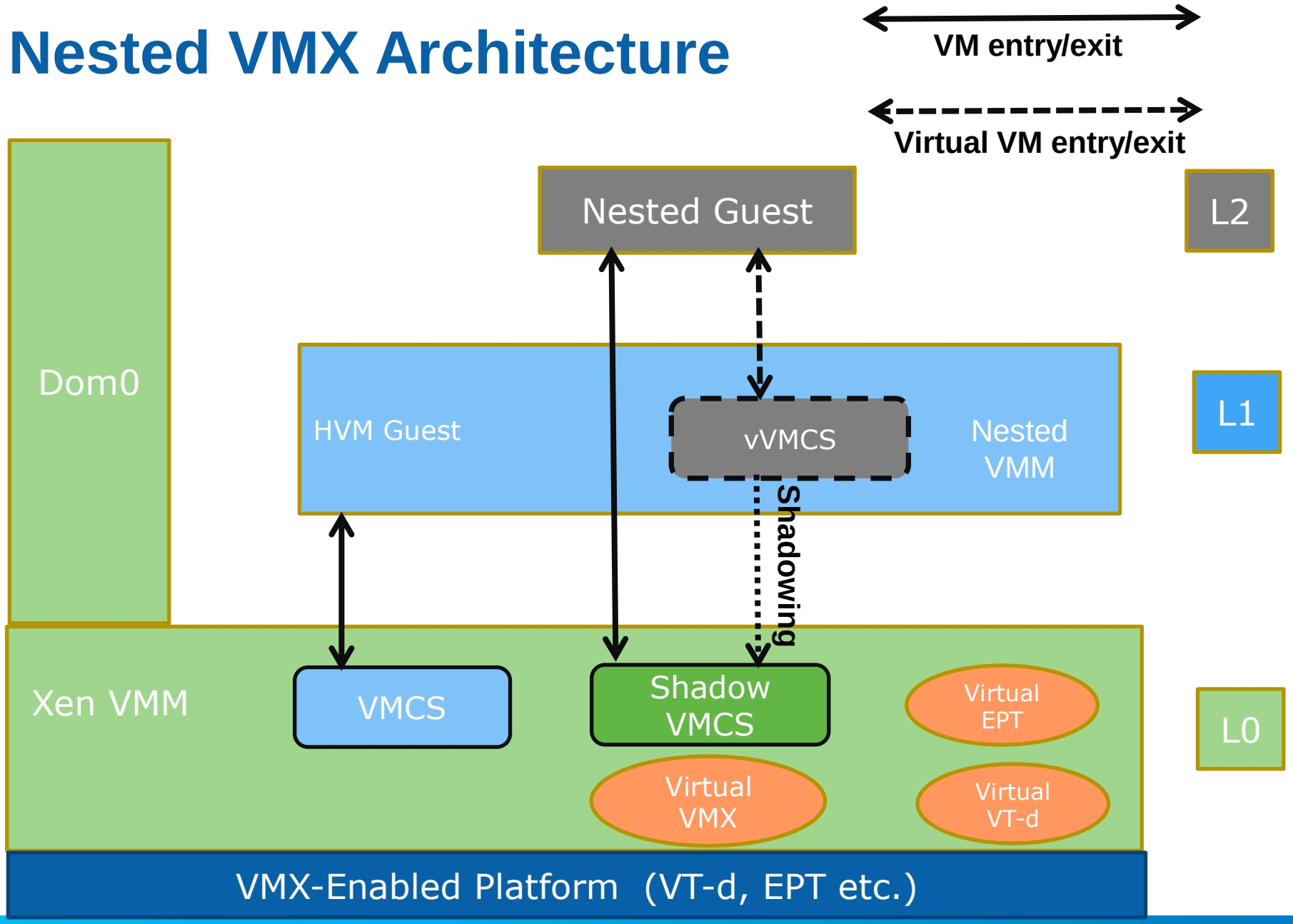
- Why nested virtualization?
  - Ordinary OS are adopting VMX now
    - Windows 7 XP compatibility mode
    - Windows 8 Hyper-V
  - Other Commercial VMMs requires VMX for better performance
    - vmware vmm
  - Anti-virus **software** depends on VMX
    - McAfee Deep Defender
- What is the goal ?
  - To make VMX-based system software run smoothly in a Xen guest.



Agenda

- Motivation and Goals
- History
  - Nested VMX Architecture
  - Previous status
- Latest status and new features
  - Stability Enhancements
  - Virtual EPT
  - Virtual VT-d
- Preliminary Performance
- Call to Action

Nested VMX Architecture



History

- **Nested VMX update @ Xen Summit Asia (Nov. 2009)**
  - Nested VMX design is presented
  - Showed Initial Status
    - Nested guest can boot up to BIOS early stage with limitations
      - single vCPU/single nested guest/ No vCPU migration
- **Refined nested VMX support was pushed into upstream**
  - Support multiple nested guests
  - Also includes supporing SMP nested guests
- **However, experimental & preliminary support**
  - Very limited configurations can work
    - "KVM on Xen", Linux guest can successfully boot up
    - "Xen on Xen" does not work
  - No virtual VT-d, virtual EPT

Previous Status

- Only one combination can work

L0-VMM	L1-VMM	L2 Guest OS						
		32Bit PAE OS			64Bit OS			
		RHEL6.0	RHEL5.4	Win7	RHEL6.0	Win7	Win2012 Server	Ubuntu 12.04
Xen	Xen	X	X	X	X	X	X	X
	KVM	X	√	X	X	X	X	X

Agenda

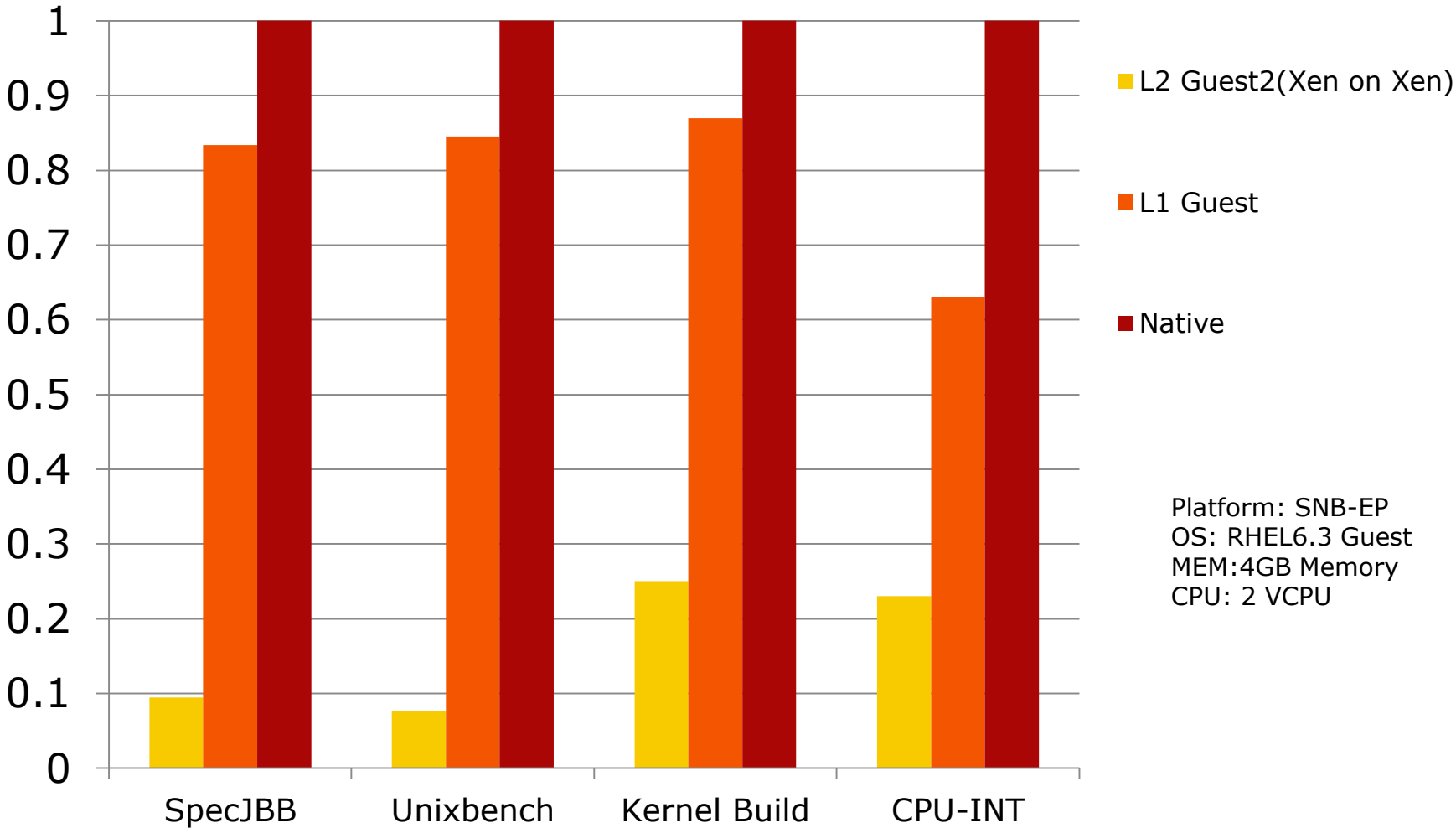
- Motivation and Goals
- History
  - Nested VMX Architecture
  - Previous status
- Latest status and new features
  - Stability Enhancements
  - Virtual EPT
  - Virtual VT-d
- Preliminary Performance
- Call to Action

Stability Enhancement

- Greatly enhanced stability, with several critical bugs fixed!

L0-VMM	L1-VMM	L2 Guest OS(SMP)						
		32Bit PAE OS			64Bit OS			
		RHEL6.0	RHEL5.4	Win7	RHEL6.0	Win7	Win2012 Server	Ubuntu 12.04
Xen	Xen	√	√	√	√	√	√	√
	KVM	√	√	√	√	√	√	√

Performance Without Optimizations

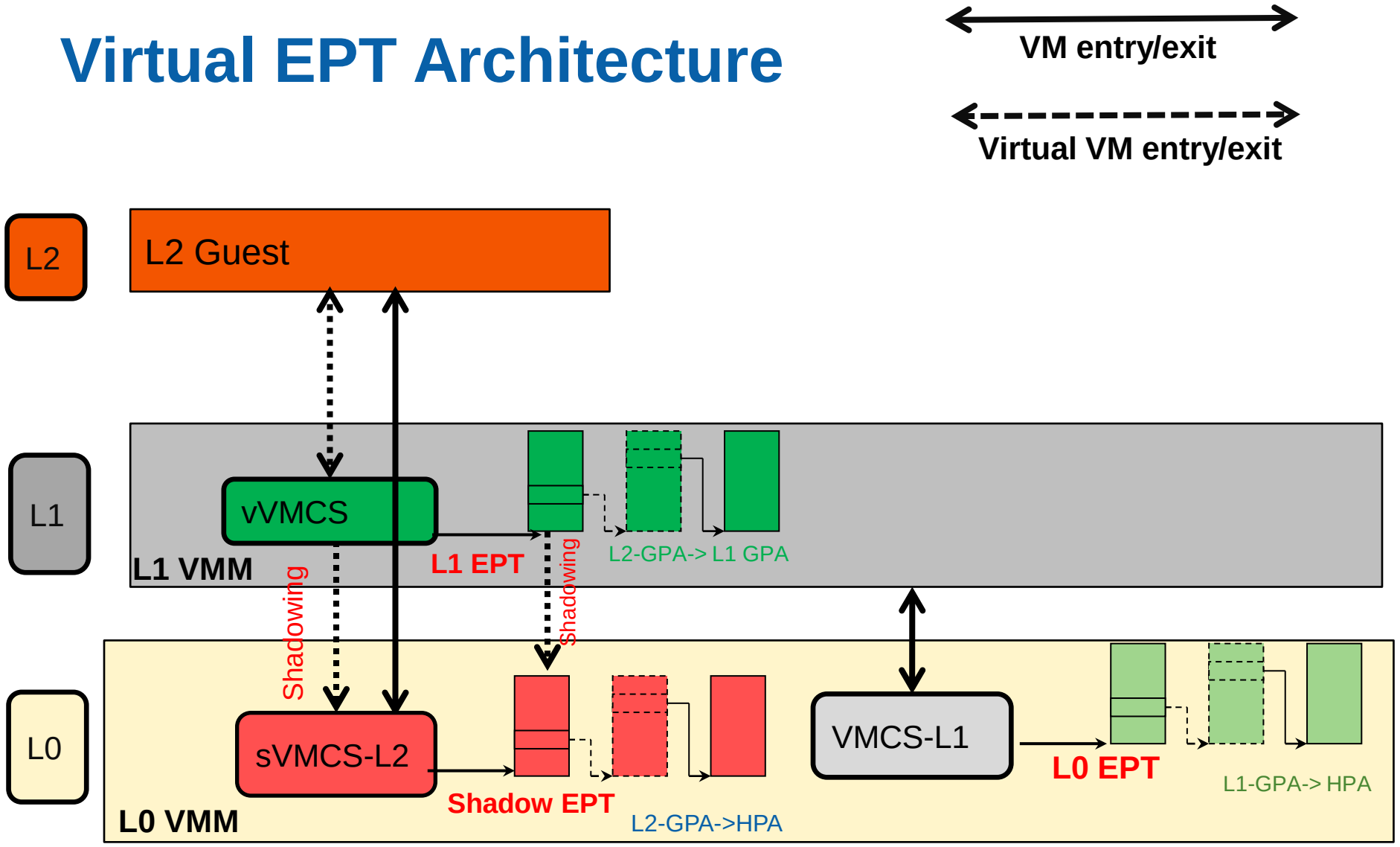


Platform: SNB-EP
OS: RHEL6.3 Guest
MEM:4GB Memory
CPU: 2 VCPU

Agenda

- Motivation and Goals
- History
  - Nested VMX Architecture
  - Previous status
- Latest status and new features
  - Stability Enhancements
  - Virtual EPT
  - Virtual VT-d
- Preliminary Performance
- Call to Action

Virtual EPT Architecture



Switch to Shadow EPT @ virtual vmentry

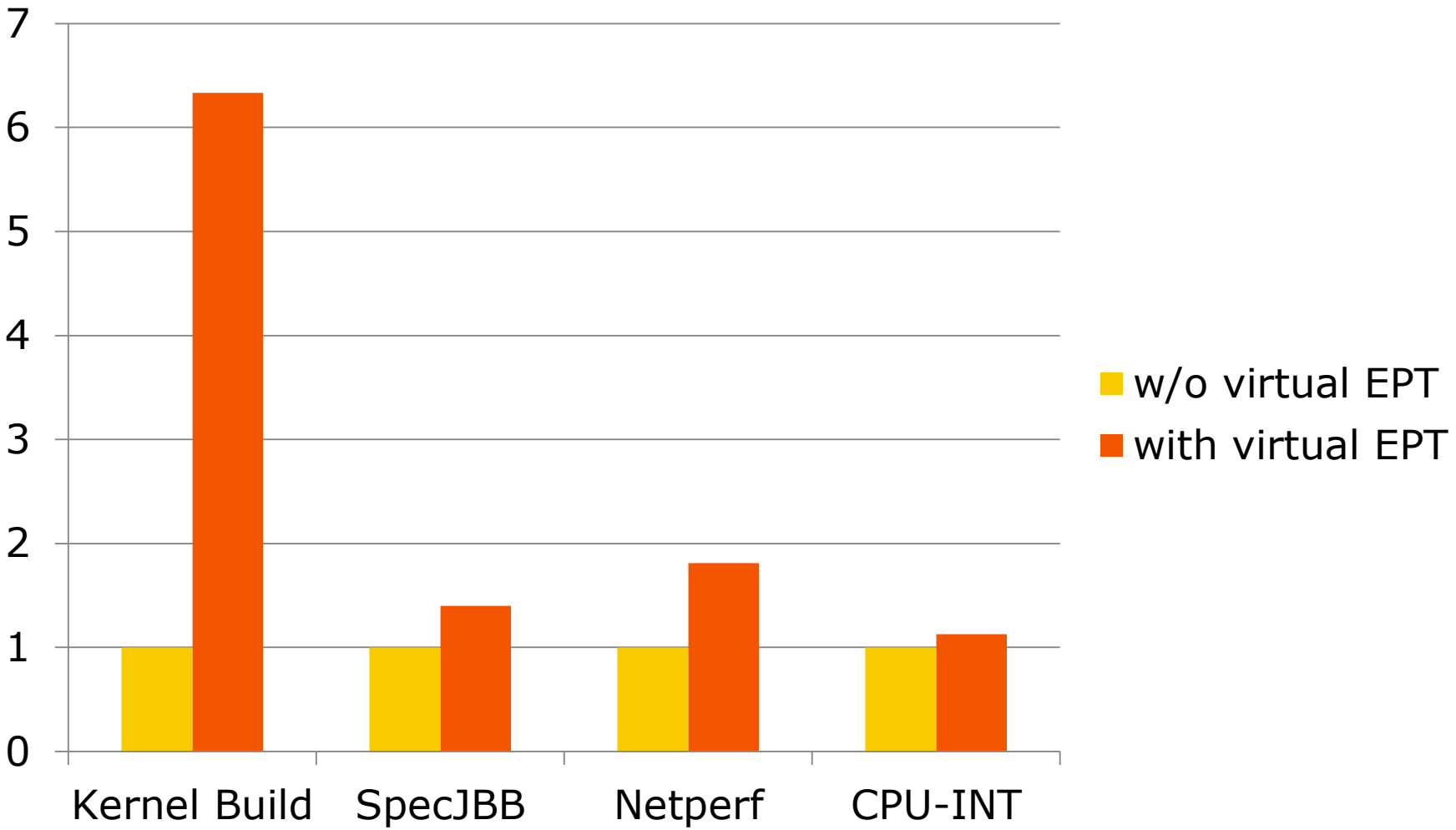
Virtual EPT: Using EPT Shadowing

- No write-protection to L1-EPT (Guest EPT paging structure)
  - Flexibility is good.
- Trap-and-emulate guest's INVEPT
  - Update the shadow EPT entries
- Better SMP Scalability
  - No global lock is required
- Requires page-level INVEPT
  - Individual address invalidation

Enhanced INVEPT Instruction for Virtual EPT

- **INVEPT limitations**
  - No Individual address invalidation
    - Only single context and all context invalidation
      - Little performance impact, however, hurt nested performance sharply!
    - Has to drop shadow EPT table for L1's each INVEPT(with single context)
      - Performance loss if frequent INVEPT in VMM
      - For example, KVM
- **Enhance it in Software Way**
  - Add Individual address invalidation for virtual EPT
    - Expose it to nested VMM through PV approach
  - Need to enhance VMMs
    - Easy implementation for Xen and VMM
- **Benefits**
  - Reduce frequent shadow EPT paging structure flush

Performance Evaluation For Virtual EPT



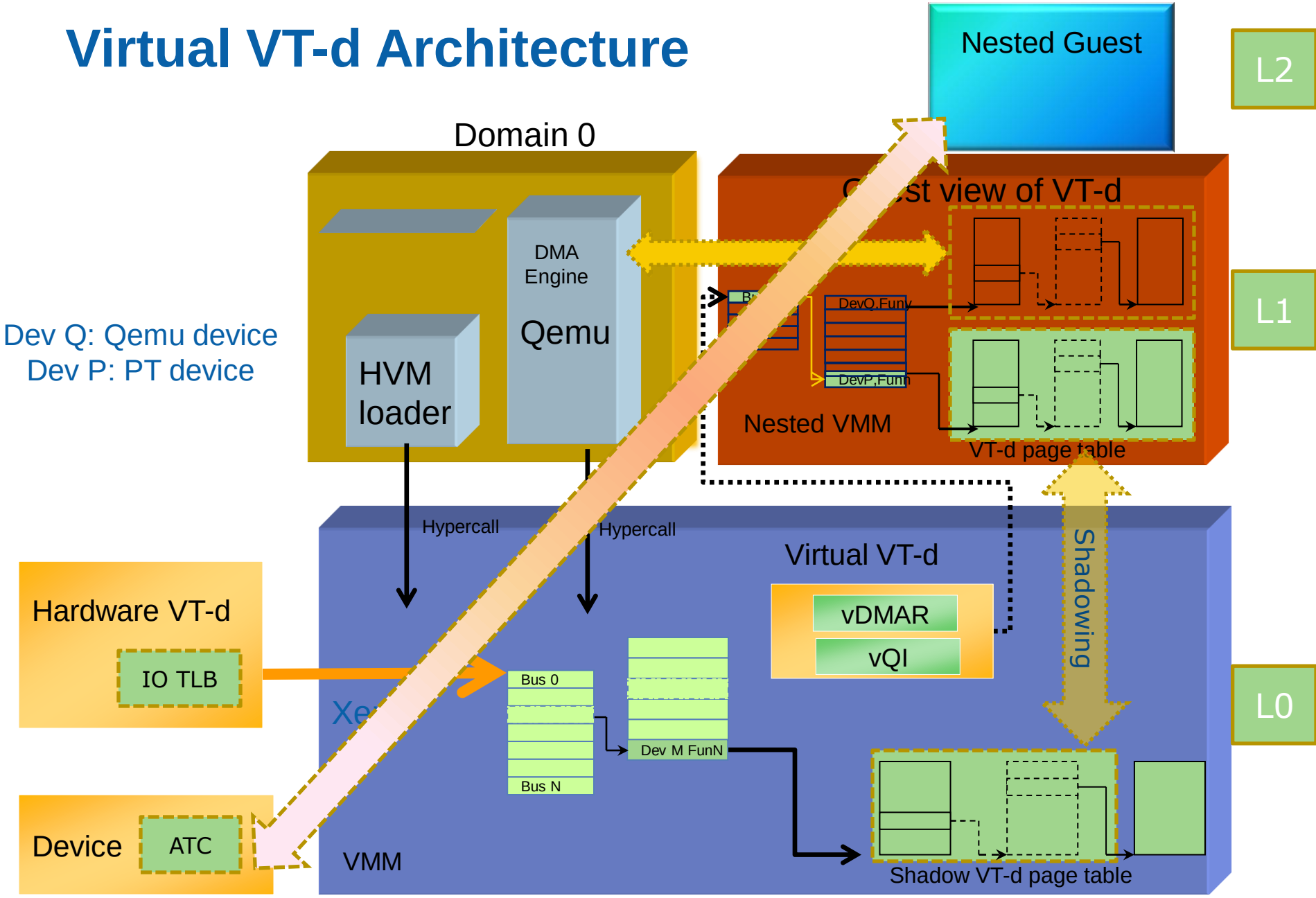
Agenda

- Motivation and Goals
- History
  - Nested VMX Architecture
  - Previous status
- Latest status and new features
  - Stability Enhancements
  - Virtual EPT
  - Virtual VT-d
- Preliminary Performance
- Call to Action

Virtual VT-d: Expose VT-d Capability to L1VMM

- **I/O performance for L2 guest is very slow**
  - Due to extremely long device emulation path through all the way to L1 & L0 VMMs
- **How to fix that?**
  - Present virtual VT-d engine to L1 VMM
  - So, device can be directly assigned to L2 guest
    - High I/O performance, because of minimum VMM intervention.
- **Must-to-have features in Virtual VT-d**
  - DMA Remapping & Queue Invalidation: Exposed
  - Interrupt remapping: Not Exposed

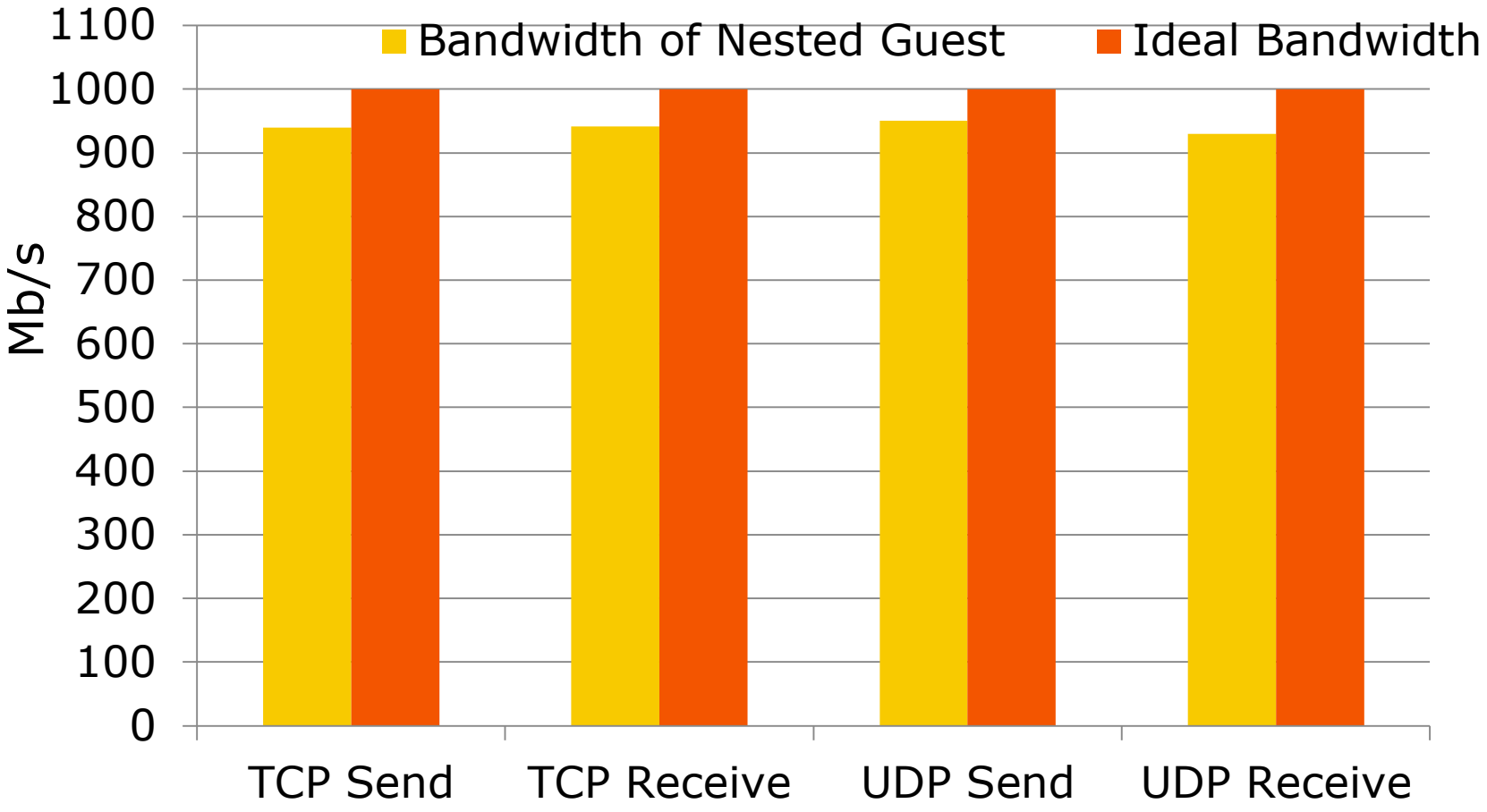
Virtual VT-d Architecture



Two types of guest devices

- **Pass through device**
  - DMA (IOVA->GPA) is handled by hardware VT-d engine
    - Remap guest root/context structure
    - Use physical remapping table to emulate guest remapping table
      - IOVA -> L0 HPA, + audit (use a dummy page for Out of Bound gpn)
      - Maybe cached by IOTLB and ATC
  - IOTLB/Context Cache Synchronization
    - Track guest invalidation of IOTLB
      - Invalidate physical IOTLB, and may invalidate ATC as well if the device has ATC
    - Track guest invalidation of Context Cache
- **Qemu device**
  - DMA (IOVA->PA remapping) is emulated by Qemu
    - 2 Options: Caching the remapping table, or No-Caching
  - Starting from simple solution: No caching
    - Qemu device is already slow

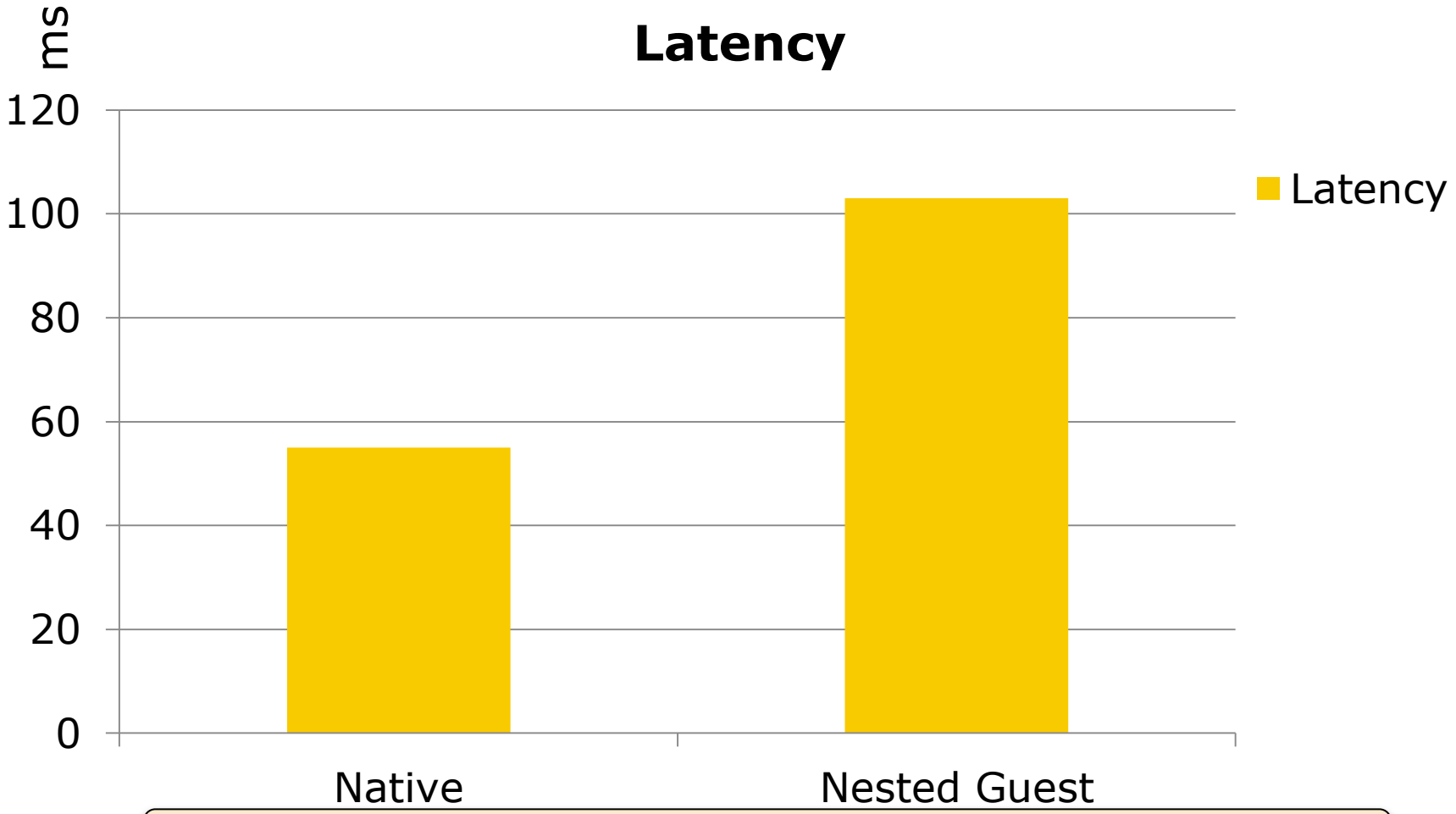
Performance Evaluation of virtual VT-d



Iperf testing with the assigned NIC to nested Guest

Bandwidth is good enough!

Latency Evaluation of virtual VT-d



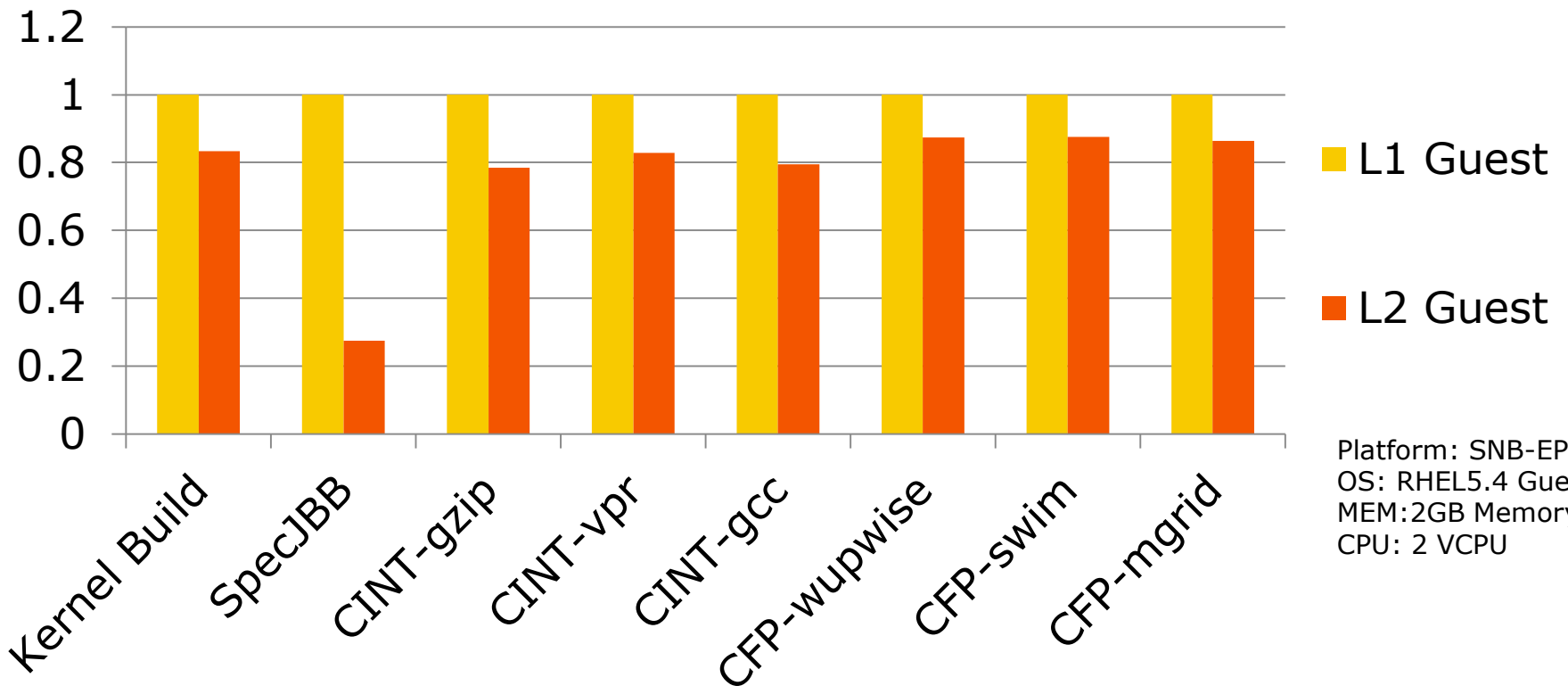
Still have room to tune Latency

Agenda

- Motivation and Goals
- History
  - Nested VMX Architecture
  - Previous status
- Latest status and new features
  - Stability Enhancements
  - Virtual EPT
  - Virtual VT-d
- Preliminary Performance
- Call to Action

Preliminary Performance

Based on Xen #25467



Platform: SNB-EP
OS: RHEL5.4 Guest
MEM:2GB Memory
CPU: 2 VCPU

Agenda

- Motivation and Goals
- History
  - Nested VMX Architecture
  - Previous status
- Latest status and new features
  - Stability Enhancements
  - Virtual EPT
  - Virtual VT-d
- Preliminary Performance
- Call to Action

Call to Action

- **Support more L1 VMMs**
  - McAfee Deep Defender
  - VMware VMM
  - Hyper-V
  - Virtual Box
- **Virtual APIC-V**
  - New Features for Interrupt/APIC Virtualization are coming
  - For more information, please come to Nakajima Jun's talk "Intel Update" this afternoon.
  - Improve interrupt virtualization efficiency for both L1 and L2
- **Performance Tuning**

Reference

- **Nested Virtualization on Xen**
  - Qing He:
  - Xen Summit 2009: http://xen.org/xensummit/xensummit_fall_2009.html
- **Virtual APIC-V**
  - Jun Nakajima: Intel Update
  - Xen Summit 2012: http://www.xen.org/xensummit/xs12na_talks/T10.html

Questions?

- **Or contact xiantao.zhang@intel.com**