

# Optimizing and Enhancing VM for the Cloud Computing Era

20 November 2009

Jun Nakajima,  
Sheng Yang, and Eddie Dong

# Implications of Cloud Computing to Virtualization

- More computation and data processing
  - More Memory
  - More CPU
    - Grid, Cluster, Parallel Computing, HPC
- More I/O activities
  - Network, Storage
- More security requirements
  - Encryption, etc.
- More VMs and power consumption

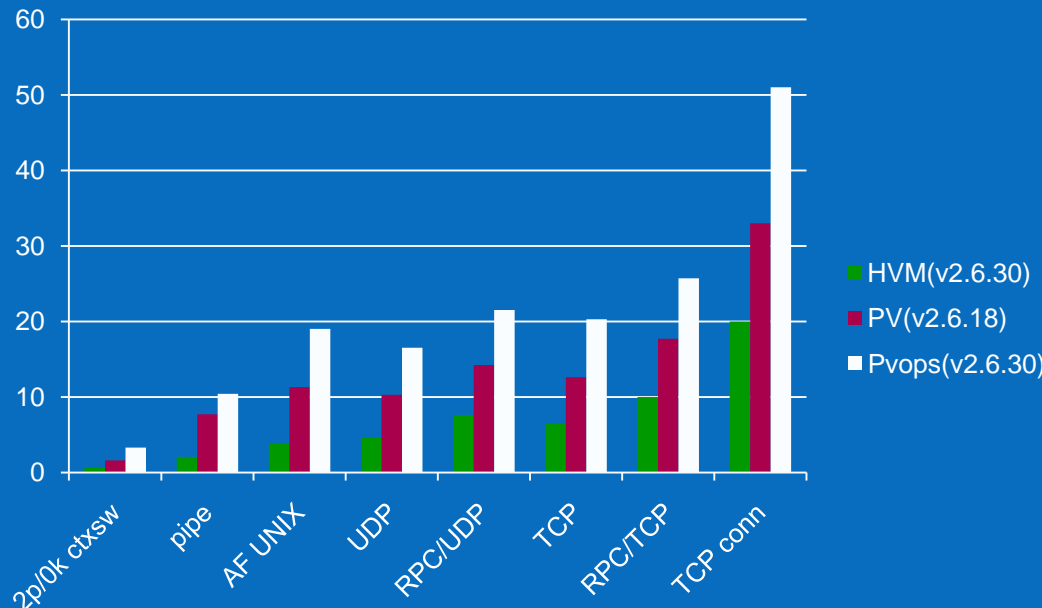
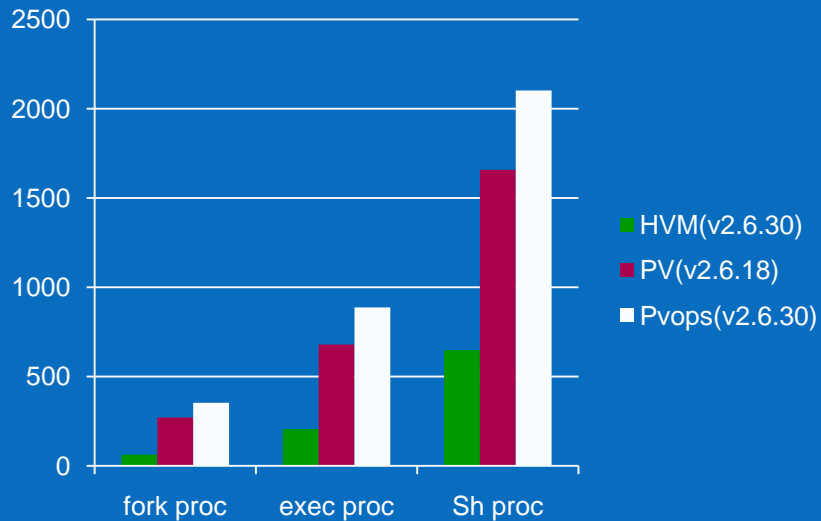
Continue to minimize virtualization overheads and utilize H/W features as much as possible

# Issues and Proposal

- 64-bit HVM (Hardware-based Virtual Machine) is at par or faster than 64-bit PV (Para-Virtualized) VM on new machines in market
  - Ring0 kernel, Hardware Assisted Paging (HAP, e.g. EPT), Lower VM exit/entry cost
    - More with Direct I/O (e.g. VT-d) and SR-IOV
    - Future processors all have those
  - More H/W virtualization optimizations/features are coming in the future
- Time to think about new VM architecture for Xen
  - Xen para-virtualization is helpful for HVM guests
    - Simplicity, performance, efficiency, scalability, correctness
  - Dom0 performance should be improved if we run it in HVM on new machines
- Combine HVM and Para-Virtualization taking advantage of both
  - “Hybrid Virtualization” ( $\geq$  HVM,  $\geq$  PV)

Jun Nakajima and Asit K. Mallick, *Hybrid-Virtualization—Enhanced Virtualization for Linux*, in *Proceedings of the Linux Symposium*, Ottawa, June 2007

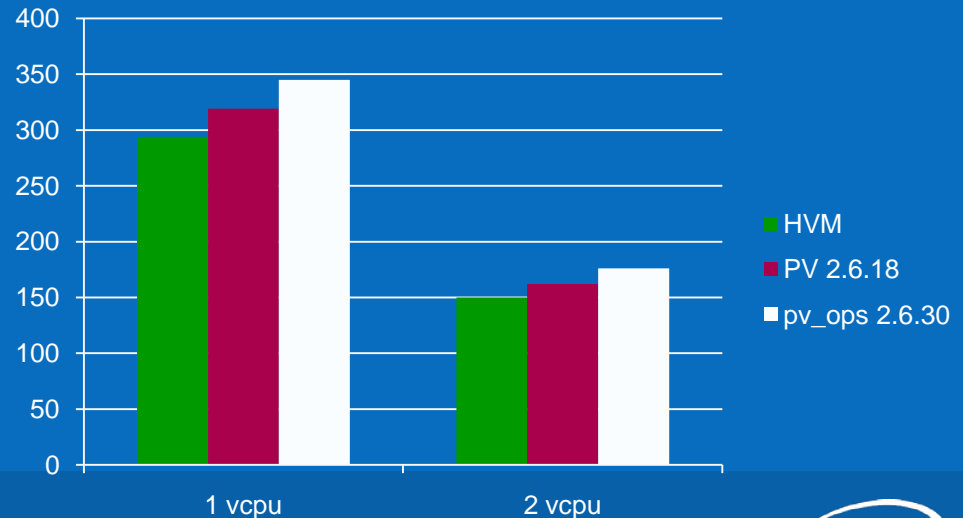
# PV Linux vs. HVM Linux



The smaller is the better

- Imbench (above)
- Kernel build (lower right)

Performance regression with pv\_ops PV Linux



# Two Architectures of Hybrid Virtualization

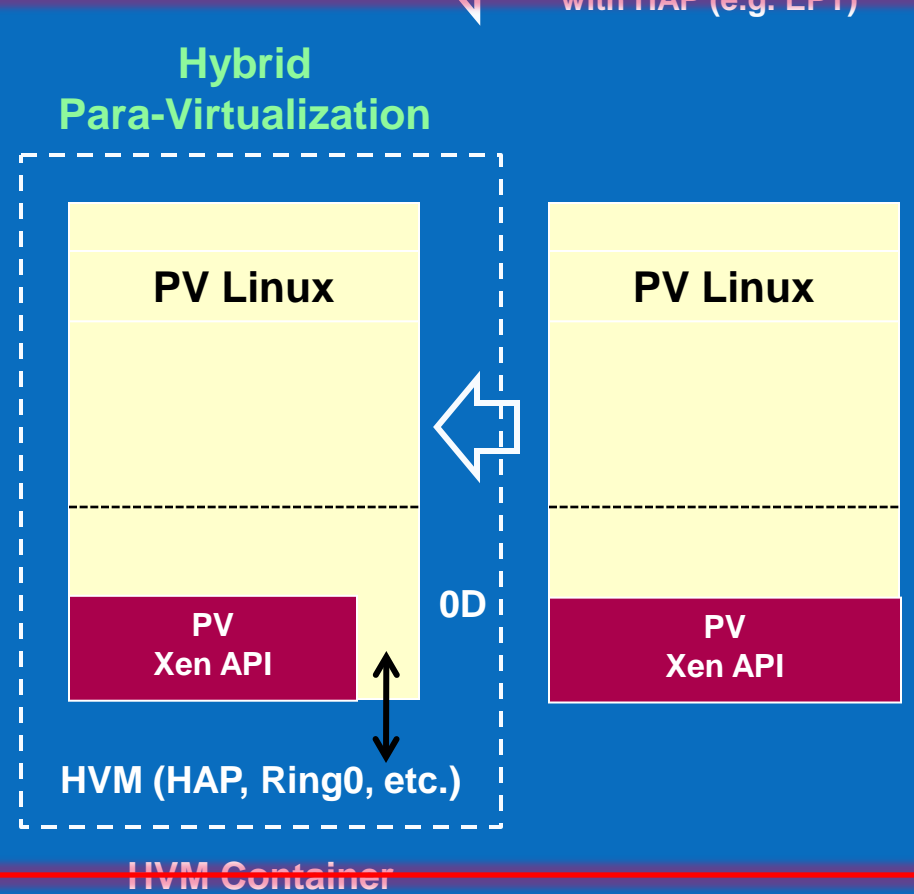
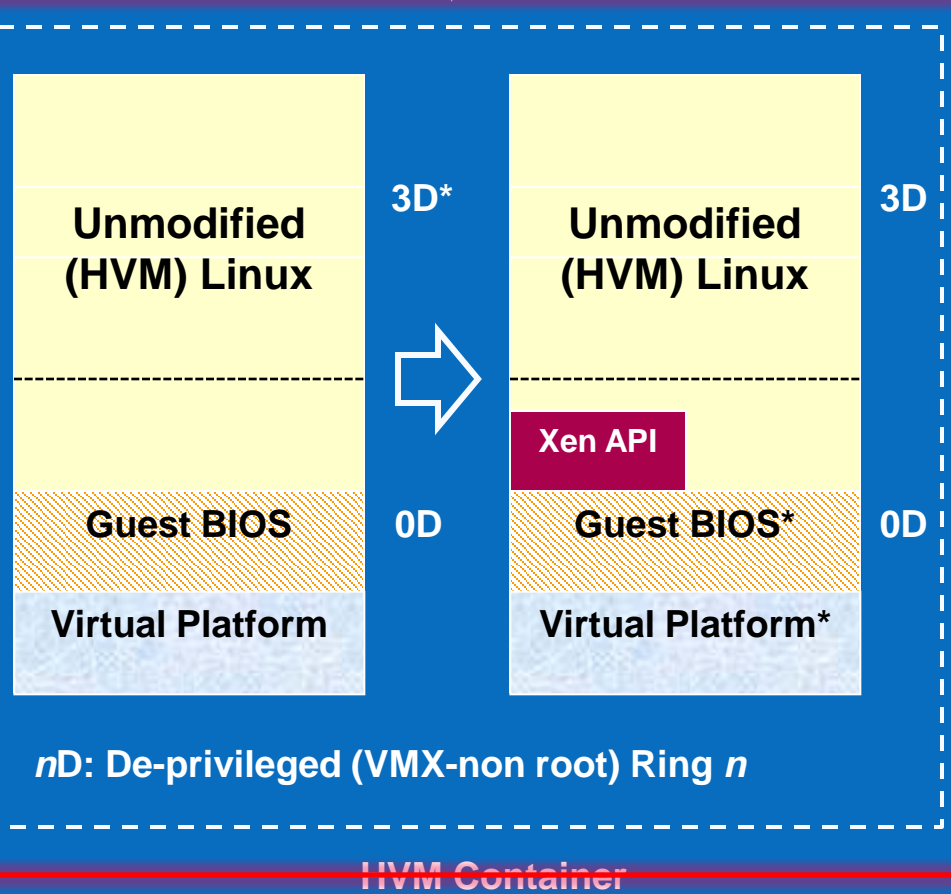
Enabled in virtualization on Xen



Hybrid HVM



Activated on machines with HAP (e.g. EPT)



\*: Native AC

Single binary can support HVM, PV, and Hybrid HVM with pv\_ops



# Hybrid PV vs. Hybrid HVM

- Hybrid PV

Pros:

- Does not require qemu-dm. Very quick to boot.
- Can be more aggressive in terms of para-virtualization

Cons:

- Does not leverage native boot features
- More changes required (compared with hybrid HVM)

- Hybrid HVM

Pros:

- Leverage native code, superset of native. Easier to re-use H/W virtualization features.
- Incremental approach (use Xen API as needed), fewer modifications

Cons:

- HVM can be slower on old machines
  - But fall back to PV anyway
- Need to modify code for native

Hybrid HVM is better option

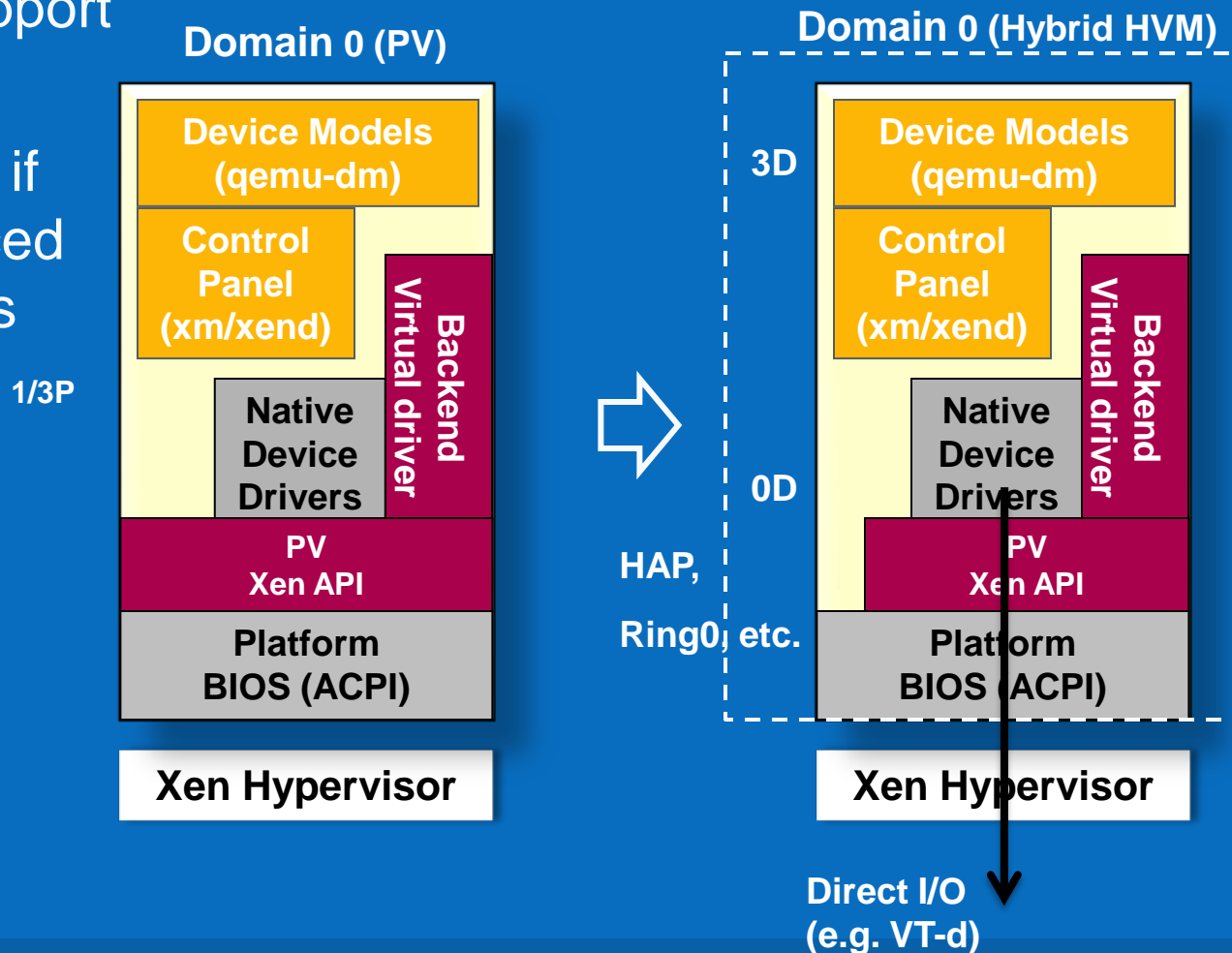
# HVM Linux vs. Hybrid HVM Linux

## Example:

- Local APIC is not used by Hybrid HVM Linux
  - EOI (End of Interrupt) does not cause VM exit.
  - Use Xen API (event channel)
- MSI, MSI-X handling is para-virtualized in Hybrid HVM Linux
  - MSI Mask/Unmask does not cause VM exit
  - No changes are made to device drivers
- I/O intensive loads expose those virtualization overheads
  - About 12K interrupts/per sec. (per VCPU) with 10 GbE (Intel Niantic, with Direct VT-d)
    - Interrupt modulation enabled: `InterruptThresholdRate = 8000` (default)
  - CPU utilization is >3% higher *per VCPU* on ordinary HVM Linux
    - EOI and MSI/MSI-X mask/unmask take up >60% of interrupt handling
  - As number of VCPUs increases, overhead increases...

# HVM Dom0

- Same binary can support both using pv\_ops
- Activate hybrid HVM if machine has advanced virtualization features
- HVM dom0 is not implemented yet



# Enhancing VM in Virtualization

- “Virtualization is better than real” Something impossible on native machine does not necessarily mean “impossible” in virtualization.
  - Opportunities for VM enhancements
  - Example: CPUs change upon live migration
- Issues with live migration
  - Least common set of CPU features are advertized in machine pool
  - Missing opportunities to gain performance boost (e.g. 30%)
- Notifications for live migration
  - Sent to kernel or special/performance-sensitive apps
    - “prepare for migration” to notify of changes to resources on migration
      - E.g. Apps know they will get new CPU features that can enhance performance
    - “complete migration”
      - E.g. Apps detect new CPU features for performance enhancements, and start using them

# Summary and Current Status

- Minimize virtualization overheads and utilize new/advanced H/W features in VMs as much as possible for cloud computing
- Hybrid HVM is superset of HVM
  - Optimized in virtualization, reduce virtualization overheads
  - Allows to extend VM for virtualization
  - Same binary for native (HVM), PV Linux, and Hybrid HVM with pv\_ops
- Prototyped Hybrid HVM
  - Support SMP, MSI/MSI-X
  - Patches sent out for comments