
ORACLE®

Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Oracle Training Materials – Usage Agreement

Use of this Site (“Site”) or Materials constitutes agreement with the following terms and conditions:

1. Oracle Corporation (“Oracle”) is pleased to allow its business partner (“Partner”) to download and copy the information, documents, and the online training courses (collectively, “Materials") found on this Site. The use of the Materials is restricted to the non-commercial, internal training of the Partner’s employees only. The Materials may not be used for training, promotion, or sales to customers or other partners or third parties.

2. All the Materials are trademarks of Oracle and are proprietary information of Oracle. Partner or other third party at no time has any right to resell, redistribute or create derivative works from the Materials.

3. Oracle disclaims any warranties or representations as to the accuracy or completeness of any Materials. Materials are provided "as is" without warranty of any kind, either express or implied, including without limitation warranties of merchantability, fitness for a particular purpose, and non-infringement.

4. Under no circumstances shall Oracle or the Oracle Authorized Boot Camp Training Partner be liable for any loss, damage, liability or expense incurred or suffered which is claimed to have resulted from use of this Site of Materials. As a condition of use of the Materials, Partner agrees to indemnify Oracle from and against any and all actions, claims, losses, damages, liabilities and expenses (including reasonable attorneys' fees) arising out of Partner’s use of the Materials.

5. Reference materials including but not limited to those identified in the Boot Camp manifest can not be redistributed in any format without Oracle written consent.

ORACLE VM

ORACLE

Oracle VM 3 – Virtual Machines

Presenter's Name
Presenter's Title

ORACLE PARTNERNETWORK

Specialized. Recognized by Oracle.
Preferred by Customers.

Planning Guest Type:
Paravirtualized Virtual Machines (PVM)
vs.
Hardware Virtualized Machines (HVM)

Performance Factors – HVM vs. PVM

- Key contributors to VM overhead:
  - Memory management
  - Timer management
  - I/O management

Anatomy of an Oracle VM Server

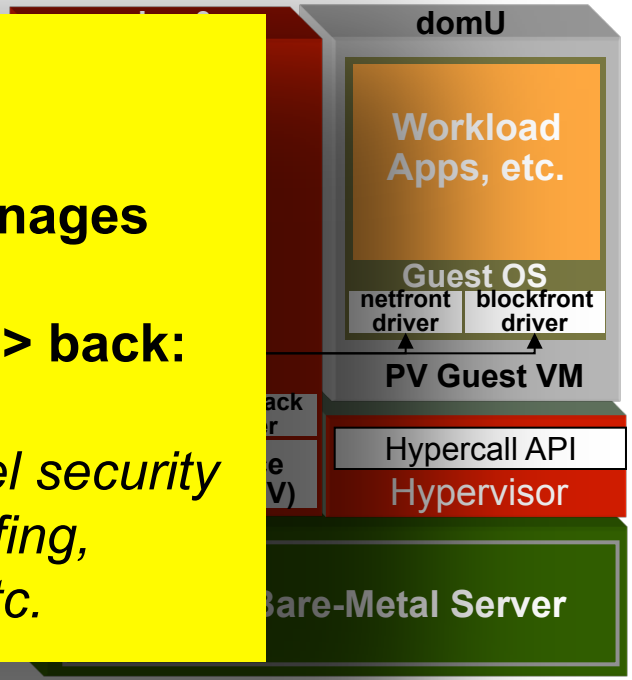
Key Concepts: Driver Architecture - PVM

- **For PV guest**
  - **Replaces ha**
    - One netw
    - One block
  - **Very stable**
    - Excellent
  - **Front-end d**
    - Inside the
  - **Back-end dr**
    - In dom0 /
  - **Open, native hardware vendor drivers**
    - Uses open Linux drivers

Paravirtualization:

Virtualization server manages high-performance communication front <=> back:

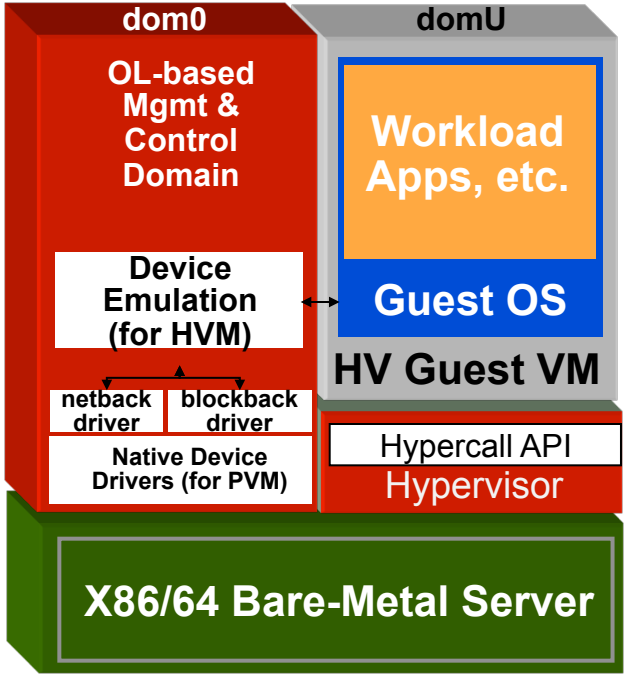
Can leverage dom0 kernel security features, e.g., packet sniffing, firewalling, rate control, etc.



Anatomy of an Oracle VM Server

Key Concepts: Driver Architecture - HVM

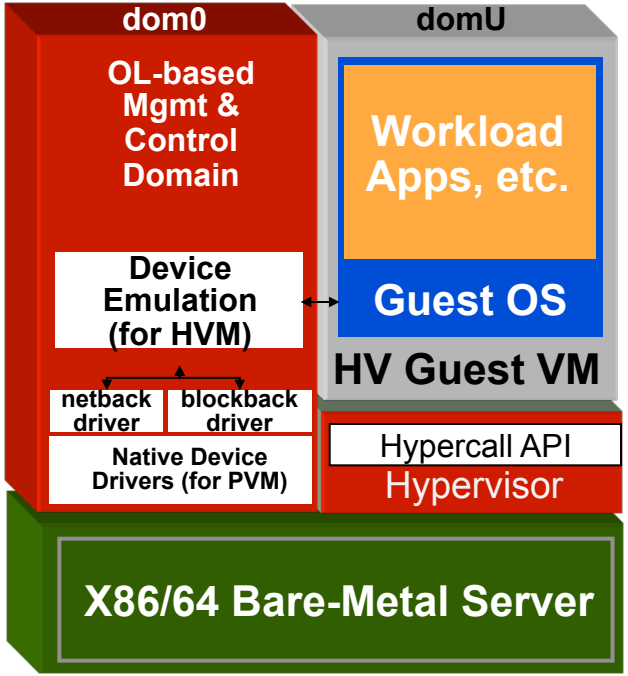
- HVM guests driver choices:
  - **Choice 1:**
  - **Unmodified native driver(s)**
    - OS typically installs basic native network and block drivers that come with the OS (not with the virtualization server)
    - Device support provided via device emulation / translation in the virtualization server on top of net- & block-back drivers in dom0
    - Slower than PV due to emulation overhead



Anatomy of an Oracle VM Server

Key Concepts: Driver Architecture - HVM

- HVM guests driver choices:
  - **Choice 2:**
  - **PV drivers** (front & back)
    - HVM (unmodified) kernels *can* use PV drivers
    - Leverages the PV driver stack same as previously described
    - Excellent performance



Performance Factors

Memory Management Overhead

Virtualization Method	Memory Management Implications & Issues	Impact on Overhead	Implications for Performance
Paravirtualization (PVM)	• Page faults handled by guest PV kernel using hardware memory management (MMU) directly • Memory page sharing in the future	• No overhead / excellent scalability • May cause swapping in guest	• Performance equals bare metal
Hardware Virtualization (HVM)	• Page faults trapped and emulated in software	• High CPU overhead with limited scalability	• Hypervisor has no knowledge of guest workload

Performance Factors

Timer Management Overhead

Virtualization Method	Timer Management Implications & Issues	Impact on Overhead	Implications for Performance
Para-virtualization (PVM)	• All VMs get real time clock directly from the dom0 clock (not emulated) • Guest PV kernel is “tickless”	• No overhead from timers vs. bare metal • No guest clock slew	• Not a factor in scaling workload
Emulation / translation	• Hypervisor emulates one clock for each VM • Generates one interrupt per scheduler tick * #VMs (i.e., 4 VMs = 4 interrupts per scheduler tick)	• Potentially large overhead from timers vs. bare metal • 4 VMs with 1KHz kernels = 4,000 interrupts/second • Significant possibility for timer slew from overload	• Significant factor in scaling workload • Slew resulting from overload can be very troublesome

Performance Factors

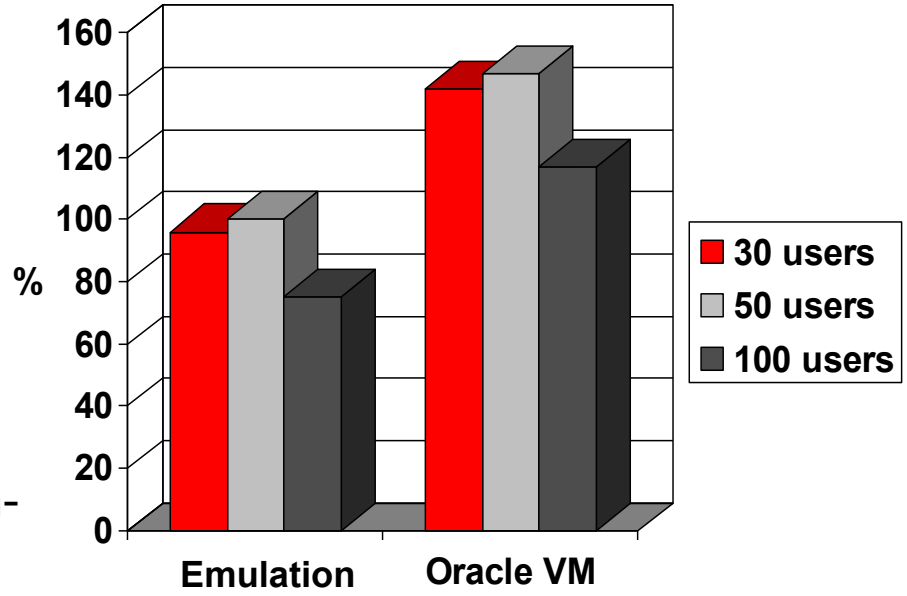
I/O Management Overhead

Virtualization Method	I/O Management Implications & Issues	Impact on Overhead	Implications for Performance
Para-virtualization (PVM)	• Thin virtual device drivers in the PV kernel communicate with back-end drivers in dom0. • Dom0 drivers are essentially normal Linux – readily available • Built into guest OS distribution	• Far less overhead vs. “real” driver for the emulated device	• Better scalability – closer to bare metal performance under load
Emulation / translation	• Must emulate I/O hardware • Proprietary drivers might not be available	• Uses the “real” device driver in the VM – adds emulation overhead	• CPU resources limit I/O throughput • Lower scalability, especially on network I/O

Relative Performance

Paravirtualization vs. Emulation

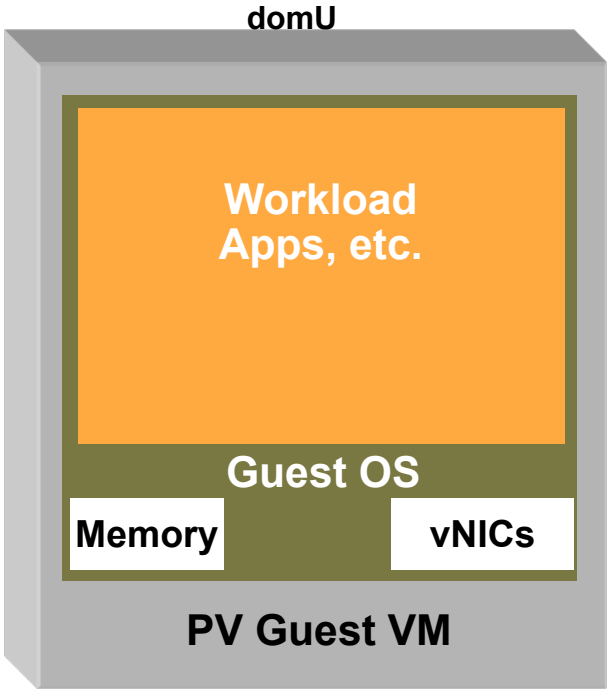
- Swingbench (DB) benchmark
- Software components
  - Enterprise Linux 4
  - Oracle 10g Database
- Physical servers
  - “Apples-to-apples”
  - 1 guest, 1GB mem, 2 vCPUs
  - Server 1: Commercial emulation-based server
  - Server 2: Oracle VM Server



Virtual Machines

Anatomy of a Virtual Machine

- Virtual Disk(s)
  - One or more
- Memory Allocation
- Virtual Network Interface(s)
- Virtual CPU(s)
- vm.cfg



Sample vm.cfg File

(1 of 3)

```
bootloader = '/usr/bin/pygrub'
vif = ['mac=00:21:f6:aa:00:25,bridge=192.168.0.0']
guest_os_type = 'linux'
name = '0004fb0000060000a16c1d729320e272'
OVM_description = ’Oracle 11g DB Server'
OVM_simple_name = 'DB1'
on_poweroff = 'destroy'
boot = 'c'
vfb = ['type=vnc,vncunused=1,vnclisten=127.0.0.1']
```

Sample vm.cfg File

(2 of 3)

```
on_crash = 'restart'
on_reboot = 'restart'
vcpus = 2
cpu_weight = 33000
OVM_os_type = 'Oracle Linux 5'
memory = 16384
cpu_cap = 0
OVM_high_availability = True
```

Sample vm.cfg File

(3 of 3)

```
disk = ['file:/OVS/Repositories/
0004fb000003000025edb8792d63ff7e/VirtualDisks/
0004fb00001200006b5af2c9371a49d0.img,xvda,w']

maxmem = 16384

uuid = '0004fb00-0006-0000-a16c-1d729320e272'

cpus = '0-3'
```

- Editing not encouraged
  - May create mismatch between VM and OVM Manager

Hard Partitioning

- Force a virtual machine to use specific physical CPUs
  - Can materially reduce software licensing costs
- cpus = ‘0,3’
  - Manually edit vm.cfg
  - Oracle VM Manager will support it in the future
- Settings in vm.cfg lost during migration
  - http://www.oracle.com/technetwork/topics/virtualization/ovm-hardpart-167739.pdf

Hardware and Software

ORACLE

Engineered to Work Together


ORACLE®