



# KVM as a Microsoft-compatible hypervisor.

Vadim Rozenfeld <[vrozenfe@redhat.com](mailto:vrozenfe@redhat.com)>  
KVM Forum, 2012

# Agenda

Microsoft Enlightenment

KVM as a conformant hypervisor

Performance Improvements



# Microsoft Enlightenment

An optimization to a guest operating system to make it aware of VM environments and tune its behavior for VMs. Enlightenments help to reduce the cost of certain operating system functions such as memory management. Enlightenments are accessed through the hypercall interface. Enlightened I/O can utilize the VMBus directly, bypassing any device emulation layer. An operating system that takes advantage of all possible enlightenments is said to be “fully enlightened.”

<http://msdn.microsoft.com/en-us/library/cc768527%28v=bts.10%29.aspx>



# Supported Windows Operating Systems

- Windows Vista
- Windows Server 2008
- Windows 7
- Windows Server 2008 R2
- Windows 8
- Windows Server 2012



# Conformant Hypervisor

## Minimal HV#1 Interfaces

- CPUID leaves 0x40000000 - 0x40000005
- Hypervisor synthetic MSRs
  - HV\_X64\_MSR\_GUEST\_OS\_ID
  - HV\_X64\_MSR\_HYPERCALL
  - HV\_X64\_MSR\_VP\_INDEX



# The Hypercall Environment

- Check that hypervisor is present
- Determine
  - Hypervisor version
  - Capabilities
  - Implementation recommendations
- Report the guest OS' identity
- Setup and enable the hypercall page



# Hypercall page

- No hypercalls

Hypercall  
Page

```
RET
MOV EDX, 0
MOV EAX, 2
```

- With hypercalls

Hypercall Page

```
RET
VM(M)CALL
```

```
int kvm_emulate_hypercall(struct kvm_vcpu *vcpu)
{
    unsigned long nr, a0, a1, a2, a3, ret;
    int r = 1;

    if (kvm_hv_hypercall_enabled(vcpu->kvm))
        return kvm_hv_hypercall(vcpu);
}
```



# Partition Reference Time Enlightenment

“hv\_reftime”

Guest:

- Windows 7
- Windows 7 SP1
- Windows Server 2008 R2
- Windows Server 2008 R2 SP1





# (Ke)QueryPerformanceCounter

- System time sources:
  - HPET
  - PM Timer
  - iTSC
  - Reference Time



# Invariant TSC

Host:

- Constant rate TSC
- HV\_X64\_MSR\_REFERENCE\_TSC MSR
- allows mapping the reference TSC page

Guest:

- RDTSC as a system time source

TSC reference Page

```
uint64_t TscOffset;  
uint64_t TscScale;  
uint32_t Res;  
uint32_t TscSequence;
```



# Reference Time Enlightenment as the fallback mechanism.

Host:

- System without invariant TSC
- HV\_X64\_MSR\_TIME\_REF\_COUNT MSR



# Guest Spin locks

“hv\_spinlocks=xxx”

HvNotifyLongSpinWait hypercall

Guest:

- used by a guest OS to notify the hypervisor that the calling virtual processor is attempting to acquire a resource that is potentially held by another virtual processor within the same partition.

Host:

- hypervisor indicates to the guest OS the number of times a spinlock acquisition should be attempted before indicating an excessive spin situation to the hypervisor



# Guest Spin locks (KfAcquireSpinLock)

- Pause-Loop Exiting

```
spin_lock:
```

```
    attempt lock_acquire;
```

```
    if fail {
```

```
        if(!spin_wait_count--) {
```

```
            HvNotifyLongSpinWait
```

```
        }
```

```
        PAUSE;
```

```
        jmp spin_lock;
```

```
    }
```



# Local APIC Virtualization

- “hv\_vapic”
- KVM provides accelerated MSR access to high usage memory mapped APIC registers.
- HV\_X64\_MSR\_EOI Accesses the APIC EOI
- HV\_X64\_MSR\_ICR Accesses the APIC ICR
- HV\_X64\_MSR\_TPR Access the APIC TPR
- APIC Assist Page



# Relaxed Timing

hv\_relaxed



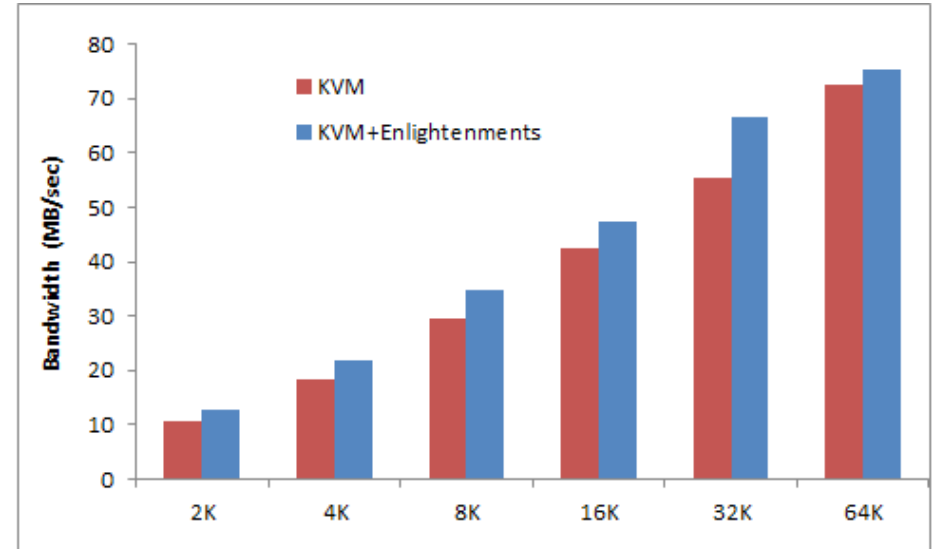
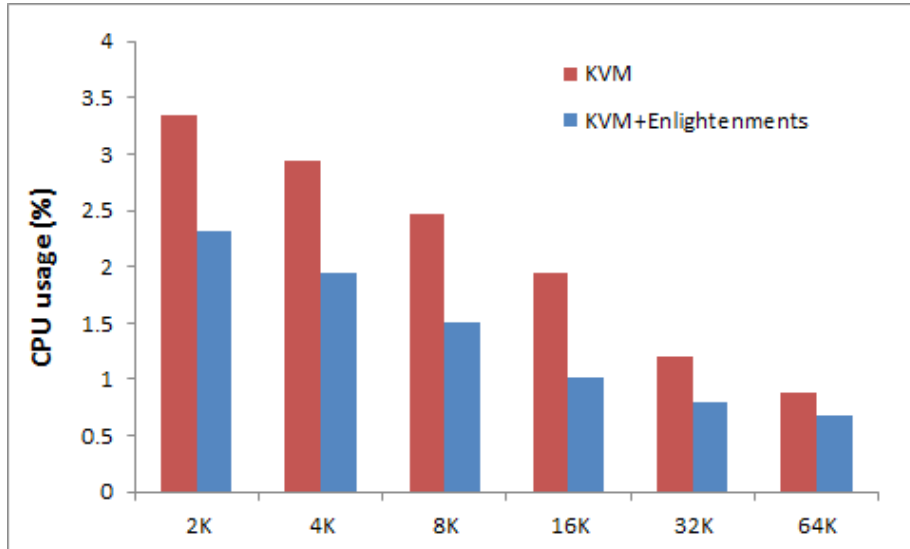
```
VNC: QEMU (vhost1365) (on usrdcor021ccxra)
A problem has been detected and windows has been shut down to prevent damage
to your computer.
A clock interrupt was not received on a secondary processor within the allocated
time interval.
If this is the first time you've seen this Stop error screen,
restart your computer. If this screen appears again, follow
these steps:
Check to make sure any new hardware or software is properly installed.
If this is a new installation, ask your hardware or software manufacturer
for any windows updates you might need.
If problems continue, disable or remove any newly installed hardware
or software. Disable BIOS memory options such as caching or shadowing.
If you need to use Safe Mode to remove or disable components, restart
your computer, press F8 to select Advanced Startup Options, and then
select safe Mode.
Technical information:
*** STOP: 0x00000101 (0x0000000000000003, 0x0000000000000000, 0xFFFFF880026CE180, 0
x0000000000000003)
```

Caused by:

- Heavy loaded.
- Interrupt delivery delays.



# IoMeter



	2K	4K	8K	16K	32K	64K
Enlgh	2.31	1.94	1.50	1.01	0.80	0.68
kvm	3.34	2.95	2.45	1.94	1.20	0.88

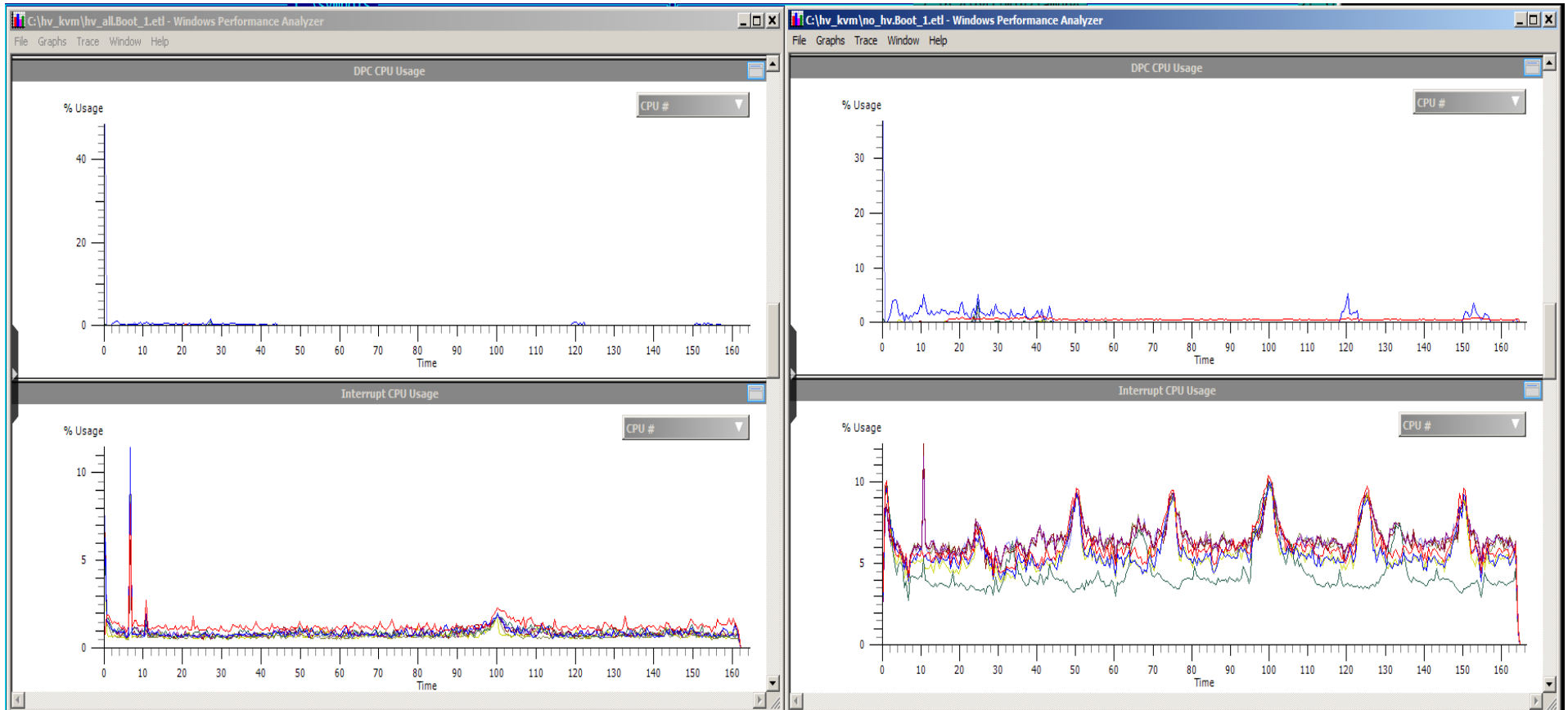
	2K	4K	8K	16K	32K	64K
Enlgh	12.80	21.87	43.69	47.46	66.38	75.21
kvm	10.51	18.28	29.64	42.50	55.49	72.52





# Windows Performance Toolkit

xbootmgr



# Viostor (virtio-blk) ISR and DPC performance

- DPC

	Max Actual Duration (ms)	Avg Actual Duration (ms)	Actual Duration (ms)
kvm + Enlgh	0.14	0.001712	19.0116
kvm	0.260369	0.011717	130.587

- ISR

	Max Actual Duration (ms)	Avg Actual Duration (ms)	Actual Duration (ms)
kvm + Enlgh	0.1841	0.002236	25.1708
kvm	0.339429	0.015927	179.8165

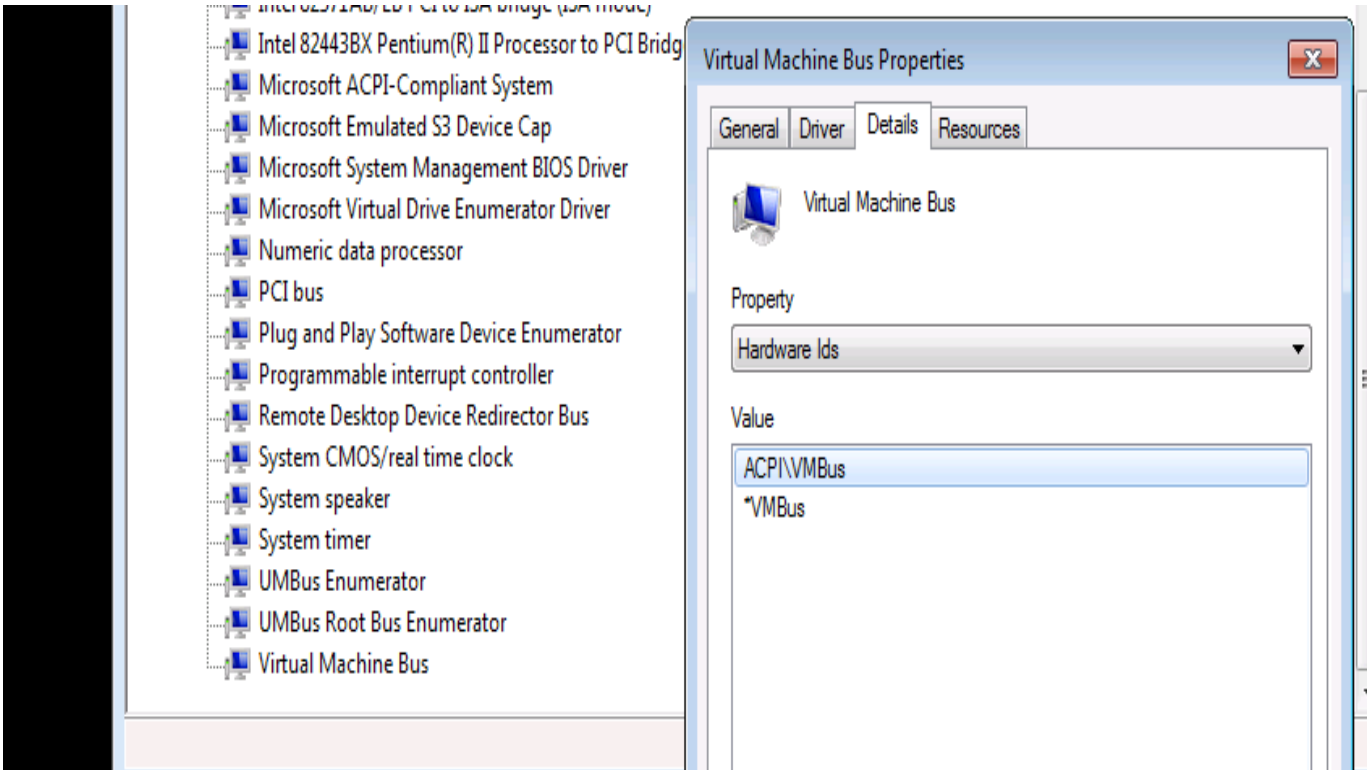


# Microsoft Hyper-V Virtual Machine Bus

“Third party virtualization solutions must not claim support for the Microsoft HyperV Virtual Machine Bus (VMBus) device in the virtual BIOS ACPI namespace.”

DefinitionBlock ("DSDT.aml", "DSDT", 1, "MSFTVM", "MSFTVM02", 0x00000002)

```
Scope (_SB.PCI0.SBRG)
{
    Device (VMBS)
    {
        Name (STA, 0x0F)
        Name (_HID, "VMBus")
        Name (_DDN, "VMBUS")
        . . . . .
    }
}
```



# Resources:

Hypervisor Top-Level Functional Specification 2.0A: Windows Server 2008 R2

<http://www.microsoft.com/en-us/download/details.aspx?id=18673>

Requirements for Implementing the Microsoft Hypervisor Interface

<http://msdn.microsoft.com/library/windows/hardware/hh975392>

