

Beyond the Hypervisor

A Technical Roadmap for Open Virtualization, Linux, KVM

Mike Day

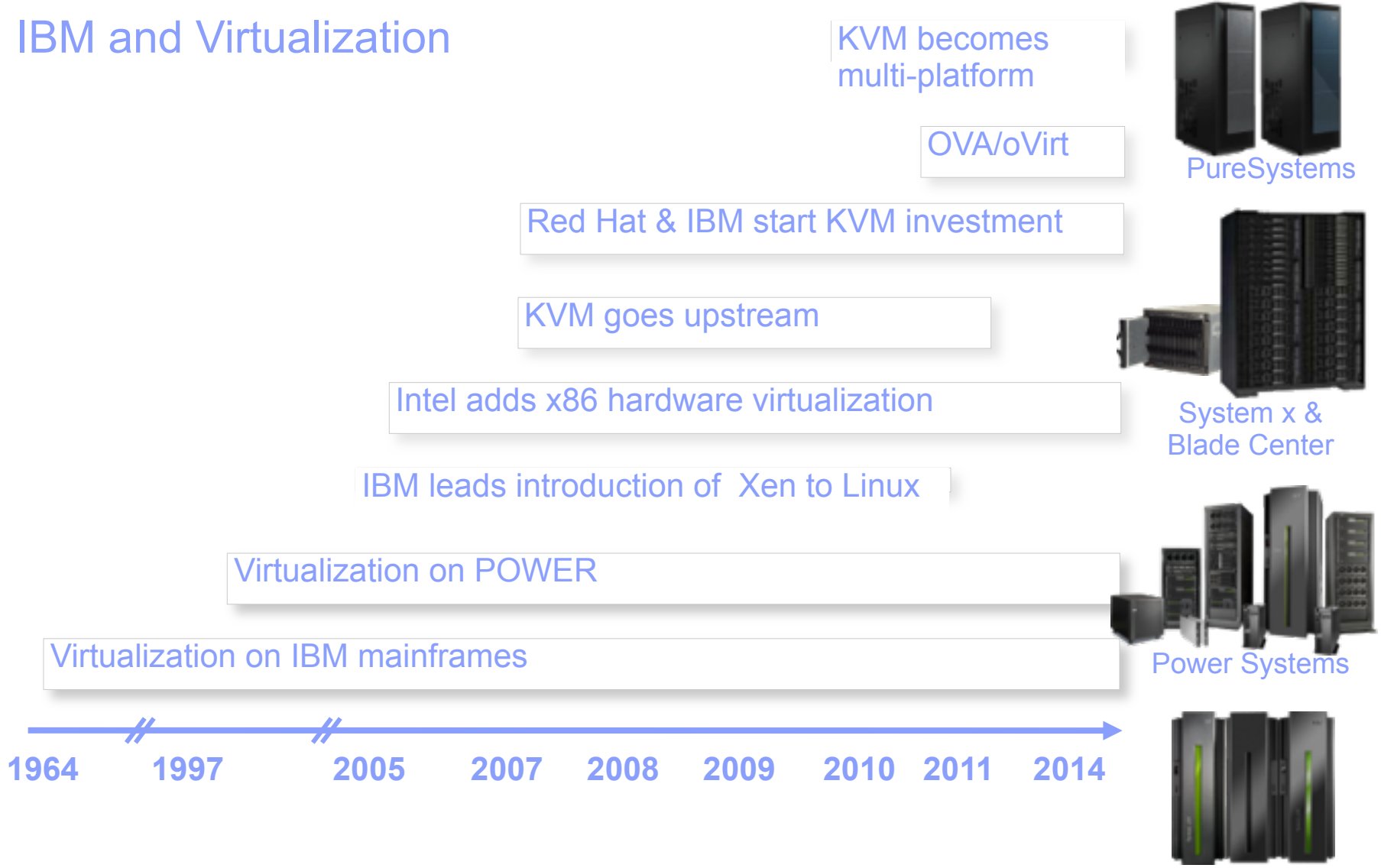
Distinguished Engineer, Chief Virtualization
Architect, Open Systems Development

Saturday, February 22, 2014



mdday@us.ibm.com

IBM and Virtualization



IBM a
A brief his



Virtual



tems



&
nter



ems



1970





1964 1997 2004-2006 2007 2008 2009 2010 2011 2014

2010



systems



x & center





systems




KVM's Unique Relationship with Linux

PUBLIC  ncultra / linux-stable

 Unwatch 1

 Star 0

 Fork 0
















branch: master linux-stable / virt / kvm

 History

Fix NULL dereference in gfn_to_hva_prot()

Gleb Natapov authored 5 months ago

latest commit a2ac07fe29

..		
 arm	ARM: KVM: Bugfix: vgic_bytemap_get_reg per cpu regs	6 months ago
 Kconfig	KVM: Introduce CONFIG_HAVE_KVM_IRQ_ROUTING	10 months ago
 assigned-dev.c	KVM: Move irq routing to generic code	10 months ago
 async_pf.c	kvm: free resources after canceling async_pf	5 months ago
 async_pf.h	KVM: Halt vcpu if page it tries to access is swapped out	3 years ago
 coalesced_mmio.c	KVM: make checks stricter in coalesced_mmio_in_range()	2 years ago
 coalesced_mmio.h	KVM: Make coalesced mmio use a device per zone	2 years ago
 eventfd.c	kvm eventfd: switch to fdget	6 months ago
 ioapic.c	KVM: Set TMR when programming ioapic entry	10 months ago
 ioapic.h	KVM: Set TMR when programming ioapic entry	10 months ago
 iodev.h	KVM: remove in_range from io devices	4 years ago
 iommu.c	kvm: Obey read-only mappings in iommu	a year ago
 irq_comm.c	KVM: Fix RTC interrupt coalescing tracking	8 months ago
 irqchip.c	KVM: Move irq routing setup to irqchip.c	10 months ago
 kvm_main.c	Fix NULL dereference in gfn_to_hva_prot()	5 months ago



KVM's Unique Relationship with Linux (cont'd.)

```

13 | {
14 |     /* vcpu data that we can't access directly from QEMU
15 |      * (i.e. with older kernels which don't support sync_regs/ONE_REG).
16 |      * Before this ioctl cpu_synchronize_state() is called in common kvm
17 |      * code (kvm-all) */
18 |     if (kvm_vcpu_ioctl(cpu, KVM_S390_INITIAL_RESET, NULL)) {
19 |         perror("Can't reset vcpu\n");
20 |     }
21 | }
22 | }

```

- **KVM Kernel Modules unlock Hardware Virtual Machine Monitor (VMM)**

- On Power (PPC64) this is a firmware VMM (level)

- **Transforms Linux into a Hypervisor**

- Kernel Engages in a Co-processing Relationship with KVM

- **KVM Uses Linux Resources**

- Scheduler, Drivers, File System, Memory Management

- **KVM Benefits from Linux Scalability, Quality, Maturity**

```

33 |
34 |     /* always save the PSW and the GPRS*/
35 |     cs->kvm_run->psw_addr = env->psw.addr;
36 |     cs->kvm_run->psw_mask = env->psw.mask;
37 |

```

Beyond The Hypervisor - Into the Kernel

- **Functions Spanning Kernel, Qemu, and Userspace**
 - **Virtual Function Input/Output (VFIO)**
 - **Brings order and relative simplicity to pci pass-through and SR-IOV**
 - **Enables GPU pass-through**
 - **NUMA Mirroring, Migration, and Control**
 - **Guest NUMA zones mirror the host's physical NUMA zones**
 - **Pin guest virtual CPUs to physical CPUs, memory nodes to physical memory nodes.**
 - **Automatic NUMA balancing and migration**
 - **IOeventfd**
 - **fast event signaling from the kernel to Qemu - for virtual interrupts and other events**

Beyond the Hypervisor - to the Data Center

- **OpenStack is Driving the KVM Roadmap**
 - **Clustered File System Integration**
 - **Distributed Block Service**
 - **Hot-plug Support for Most Resources**
 - **OpenStack convention of booting a guest with temporary resources, then hot-plugging permanent resources once the guest is on-line**
 - **Improvements in Migration Speed**
 - **Support for Network Function Virtualization**

KVM and OpenStack

- **KVM is the Choice of Over 95% of OpenStack Clouds***
 - **KVM provides the Default development environment for OpenStack**
 - **Easy to get - several reliable Linux/KVM/OpenStack distributions**
 - **Scalable, Efficient, Economical**

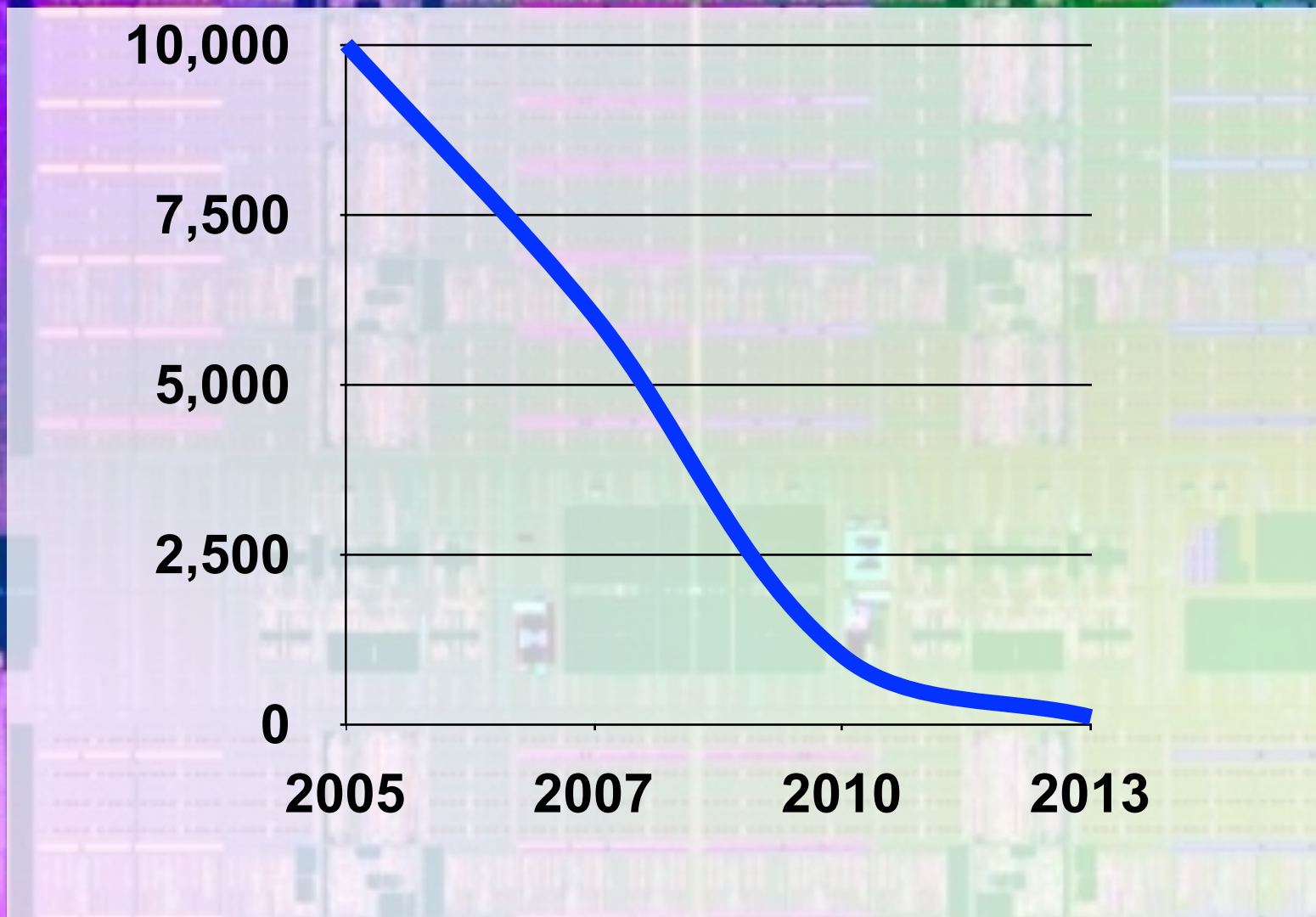
** Source: IDC White Paper, sponsored by IBM and Red Hat, KVM: Open Virtualization Becomes Enterprise Grade, February 2013*

Beyond the Hypervisor - KVM and OpenStack

- **OpenStack Obscures the Hypervisor - Which is Seen as a Benefit**
- **OpenStack Requires a Hypervisor to be fully functional**
- **This is a Great Scenario for KVM, which Fulfills the Hypervisor Role and Doesn't Get in the Way**
 - **Breaks the “Vendor Lock-in”**
- **KVM's Fast Development Cycle Keeps it Up-to-Date with New OpenStack Features**

** Source: IDC White Paper, sponsored by IBM and Red Hat, KVM: Open Virtualization Becomes Enterprise Grade, February 2013*

Beyond the Hypervisor - Into the Processor



— CPU Cycles To Execute a Bare VMExit

© 2014 IBM Corporation

Beyond the Hypervisor - Into the Processor

▪ Virtualization Functions Pulled Into Hardware Over the Years:

- Shadow VMCS - tracking of VMCS state in hardware
- Shadow Page Tables - Overtaken by Extended Page Tables (except during Migration). This reduced 90% of the VMExits in some workloads.
 - Management of Guest Virtual to Host Physical page mapping
- Timer interrupt - APIC Virtualization
 - “Tickless Mode” which reduces timer interrupts benefits virtualization performance at least as much as it reduces power consumption
- SR-IOV - self-virtualizing I/O devices.

Beyond the Hypervisor - Into the Processor

- **VMCS Shadowing**
 - “Can you run VMware as a guest?”
 - Trap expansion on vmread/vmwrite
- **The Turtles Project**
 - https://www.usenix.org/legacy/event/osdi10/tech/full_papers/Ben-Yehuda.pdf
- **Feature Validation**
- **Utility Processors - On 390**
- **Utility Processors - On x86?**
- **Today: VMCS Shadowing, APIC virtualization, Nested Virtualization**

Pushing KVM Into Advanced Workloads - High-Performance Computing

- **Advancements in Hardware Support for KVM**
- **2¹⁰ Increase in VMExit Performance since 2005**
- **160 Cores, 4 TB Per Host Supported**
- **Highest Ever SPECVirt Score - 432 VMs**
 - http://www.spec.org/virt_sc2010/results/res2011q2/virt_sc2010-20110419-00027-perf.html
- **1.5 Million IOPs within a single Guest**
 - ftp://public.dhe.ibm.com/linux/pdfs/KVM_Virtualized_IO_Performance_Paper_v2.pdf



← 1-9
WALL ST

- **Read-Copy-Update**
- **<http://en.wikipedia.org/wiki/Read-copy-update>**
- **Helped Kernel Obtain Impressive Scalability**
- **<https://github.com/bonzini/qemu/tree/rcu>**
- **Qemu In the Process of Becoming Asynchronous and More Scalable**
 - **Also Sequence Locks, Finer-grained locks**

Beyond the Hypervisor - Block I/O

- **VFIO - Virtual Function Input/Output**

- VFIO now supports setting CPU affinity on MSI interrupts.
- SCSI Devices as well as Ethernet/RDMA and Fiberchannel

- **virtio-blk-dataplane**

- **Converts Block I/O to an Asynchronous Thread**

- **ivshmem - Nahanni shared memory transport**

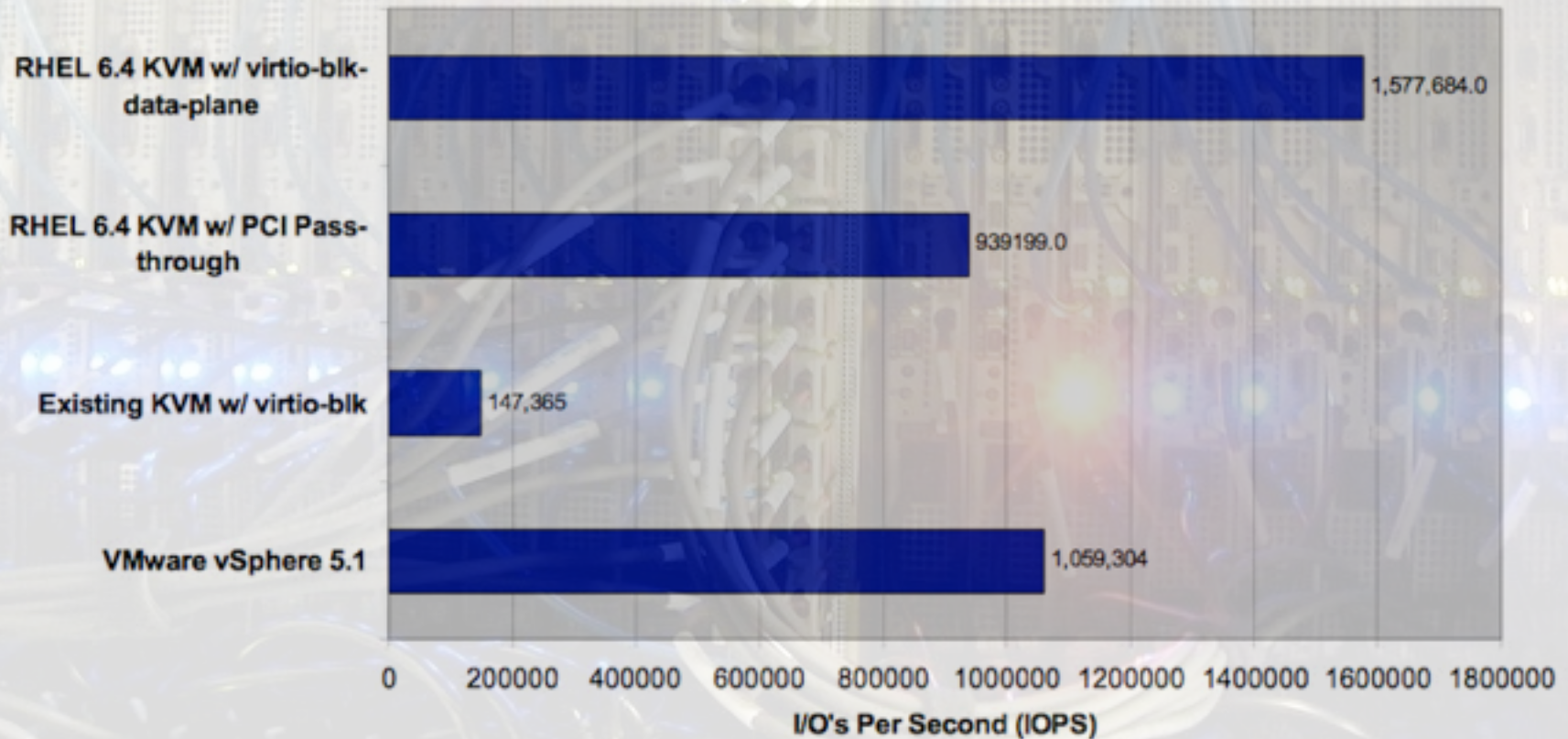
- **Win for HPC but also Applicable to Workloads that Move Bulk Data into and out of Main Memory**

- **RDMA - Remote Direct Memory Access**

- **Gluster FS - Integration, new translators**

Block I/O Performance

Single Virtual Machine Direct Random I/Os at 4KB Block Size Host Server = Intel E7-8870@2.4GHz, 40 Cores, 256GB



KVM - The First Multi-Platform Hypervisor?

```

13 | {
15 |     * vcpu data that we can't access directly from QEMU
16 |     * (i.e. with older kernels which don't support sync_regs/ONE_REG).
17 |     * Before this ioctl cpu_synchronize_state() is called in common kvm
18 |     * code (kvm-all) */
19 |     if (kvm_vcpu_ioctl(cpu, KVM_S390_INITIAL_RESET, NULL)) {
20 |         perror("Can't reset vcpu\n");
21 |     }
22 | }

```

- x86_64

```

24 |     int kvm_arch_put_registers(CPUState *cs, int level)

```

- s390

```

26 |     S390CPU *cpu = S390_CPU(cs);

```

- PPC

```

27 |     CPUS390XState *env = &cpu->env;

```

```

28 |     struct kvm_one_reg reg;

```

- ARM

```

29 |     struct kvm_sregs sregs;

```

```

30 |     struct kvm_regs regs;

```

```

31 |     int ret;

```

```

32 |     int i;

```

```

33 |

```

```

34 |     /* always save the PSW and the GPRS*/

```

```

35 |     cs->kvm_run->psw_addr = env->psw.addr;

```

```

36 |     cs->kvm_run->psw_mask = env->psw.mask;

```

```

37 |

```









Beyond the Hypervisor - Kimchi

Add Template (X)

← Remote ISO Image

The following ISOs are available:

All

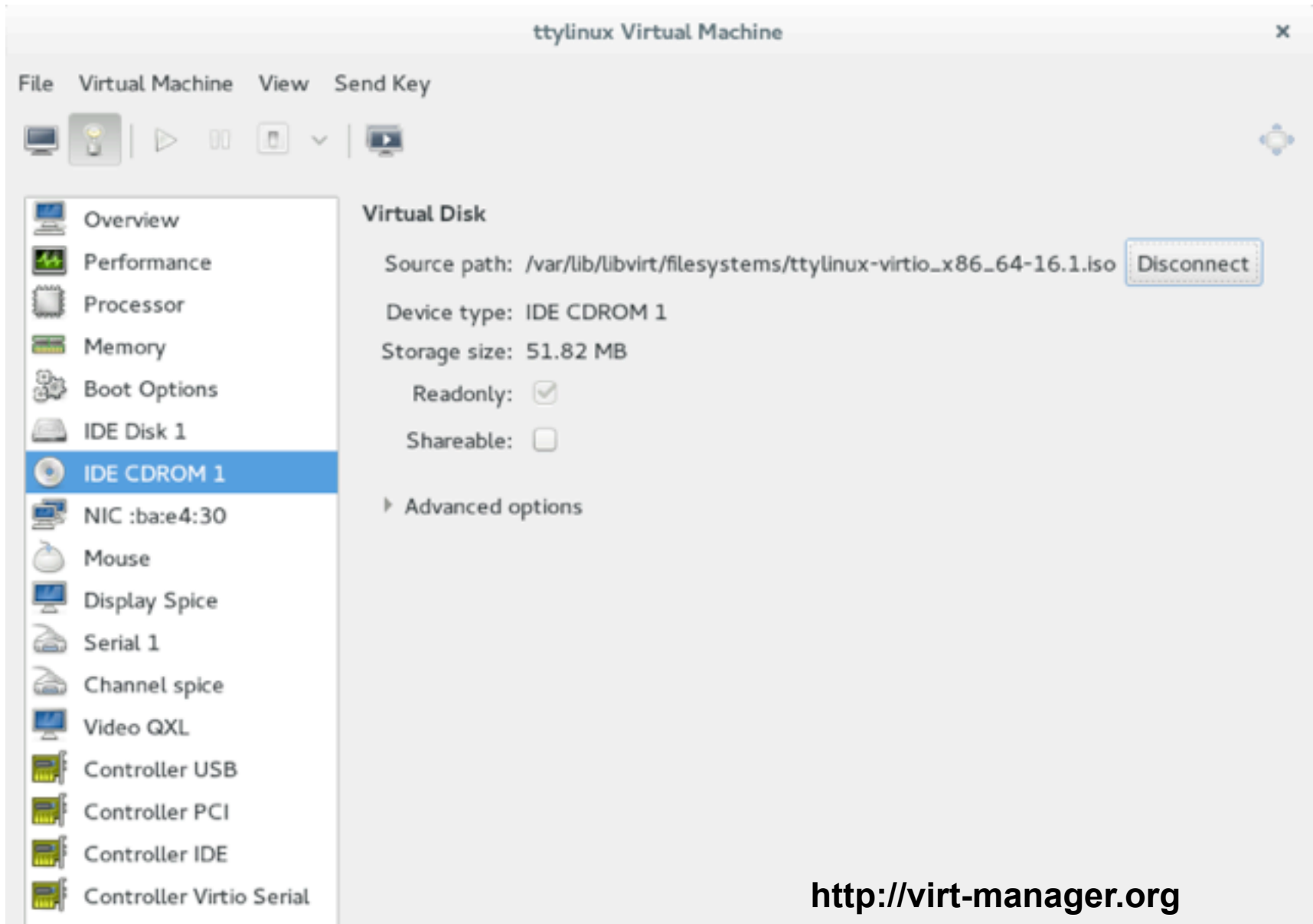
 fedora-20 OS: fedora Version: 20	 fedora-18 OS: fedora Version: 18	 fedora-19 OS: fedora Version: 19
 debian-Wheezy OS: debian Version: 7.2.0	 Ubuntu 13.10 (Saucy Salamander) OS: ubuntu Version: 13.10	 gentoo-20131010 OS: gentoo Version: 20131010
 Ubuntu 13.04 (Raring Ringtail) OS: ubuntu Version: 13.04	 opensuse-12.3 OS: opensuse Version: 12.3	

I want to use a custom URL

Create Templates from Selected ISO

<https://github.com/kimchi-project/kimchi>

Beyond the Hypervisor - Virt-Manager



Beyond the Hypervisor - oVirt

The screenshot displays the oVirt Open Virtualization Manager interface. At the top, it shows the user is logged in as 'admin@internal' and provides navigation links for 'Configure', 'Guide', 'About', and 'Sign Out'. A search bar is set to 'Host:'. Below this, a navigation menu includes 'Data Centers', 'Clusters', 'Hosts', 'Networks', 'Storage', 'Disks', 'Virtual Machines', 'Pools', 'Templates', 'Volumes', 'Users', and 'Events'. The 'Hosts' tab is active, showing a table with columns for Name, Hostname/IP, Cluster, Data Center, Status, Virtual Machines, Memory, CPU, Network, and SPM. One host is listed: 'centos-hyp01.lab.ovirt.at 10.0.100.42' in the 'ovirt-local' cluster, with 4 VMs, 75% memory usage, 1% CPU usage, and 0% network usage. Below the table, the 'Monitoring Details' tab is selected, showing a table of services and their outputs. A 'Load utilization for 10.0.100.42' graph is also visible, showing CPU load over a 4-hour period.

Name	Hostname/IP	Cluster	Data Center	Status	Virtual Machines	Memory	CPU	Network	SPM
centos-hyp01.lab.ovirt.at	10.0.100.42	ovirt-local	ovirt-local	Up	4	75%	1%	0%	SPM

Service	Output
RHEV CPU Load Check	RHEV OK: cpu ok - 1% used (centos-hyp01.l)
RHEV Host Load Check	RHEV OK: cpu.load.avg.5m ok - 0.020 (cento
RHEV Host Status Check	RHEV OK: Hosts ok - 1/1 Hosts with state Uf
RHEV KSM Load Check	RHEV CRITICAL: ksm.cpu.current critical - 90:
RHEV Memory Check	RHEV WARNING: memory warning - 75.00%
RHEV Network Status Chec	RHEV CRITICAL: Hosts critical - 1/2 Nics with
RHEV Network Traffic Check	RHEV OK: traffic ok - eth1: 0 Mbit/s eth0: 0 B
RHEV Swap Check	RHEV OK: swap ok - 19.27% used (centos-h

Load utilization for 10.0.100.42 (4 Hours)

cpu.load.avg.5m last: 0.031 max: 0.138 average: 0.07794

<http://www.ovirt.org>



www.ibm.com/systems/kvm

mdday@us.ibm.com