



The KVM Weather Report

Jes Sorensen <Jes.Sorensen@redhat.com>
LinuxCon Japan, June 1st 2011

Agenda

Virtualization Overview

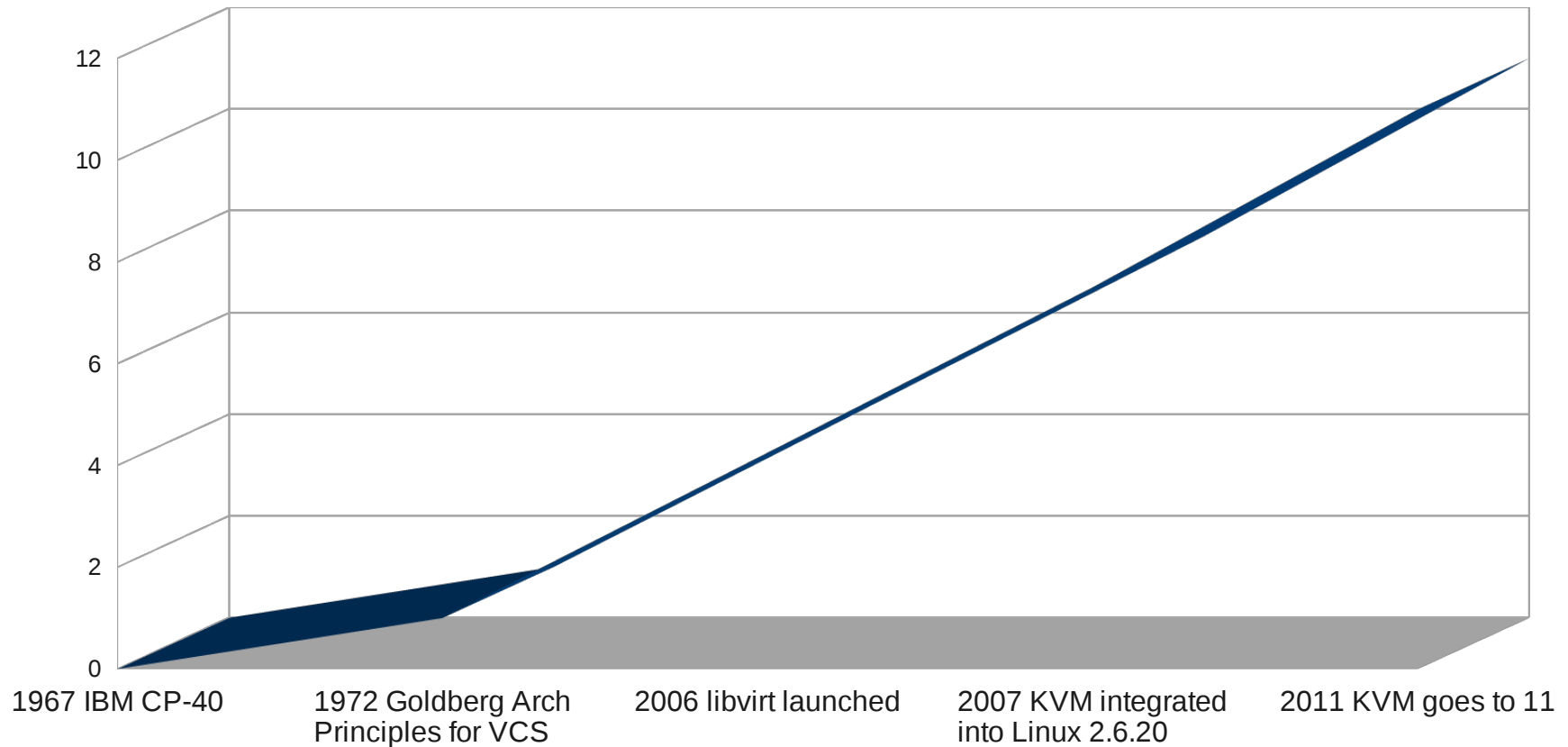
Foundation for Cloud

Future



Virtualization Overview: History

KVM goes to 11



Virtualization Overview: History

into our definition of a virtual computer system in order to distinguish it from a number of other objects which have often been casually called virtual machines.

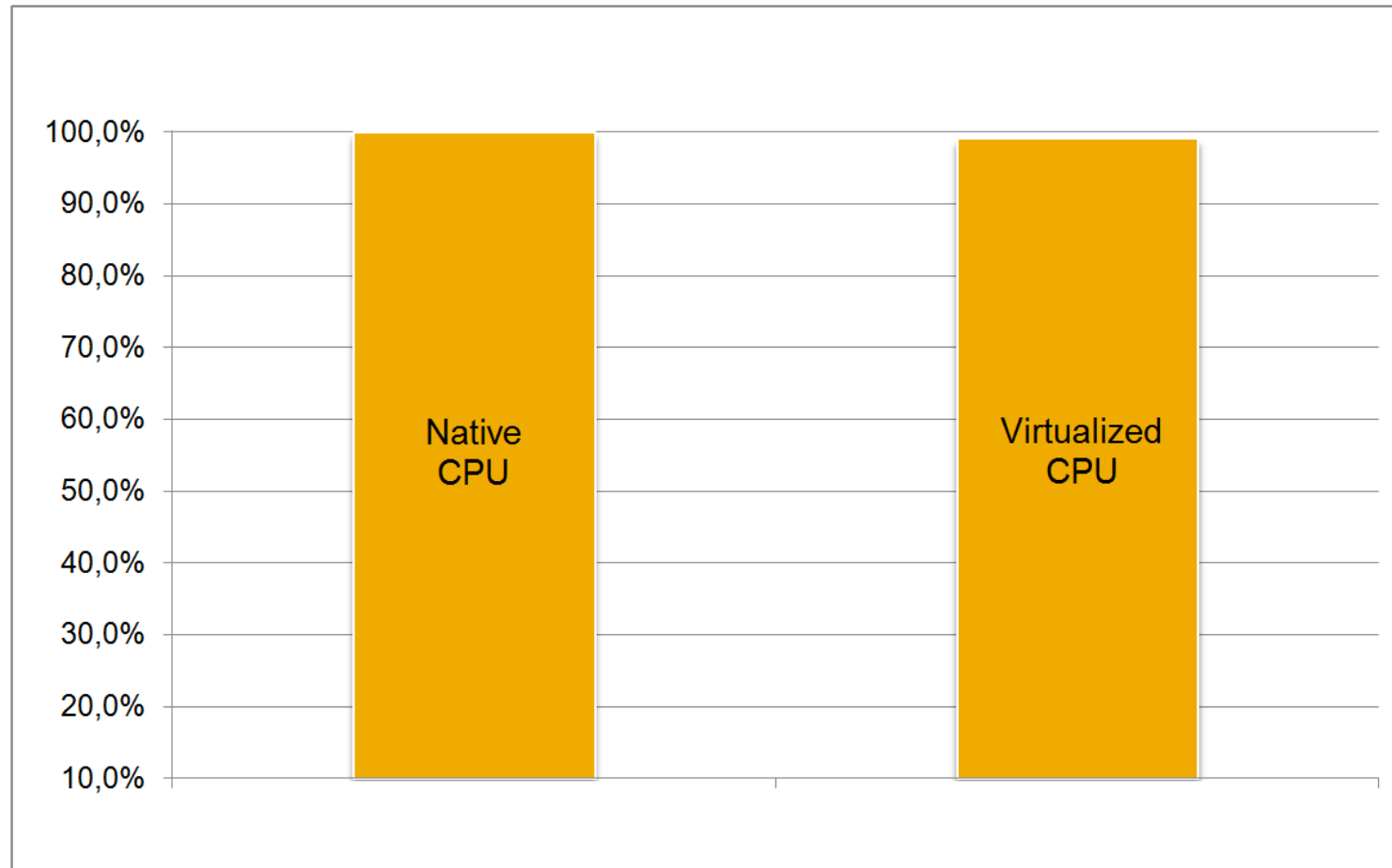
A virtual computer system is a hardware-software duplicate of a real existing computer system in which a statistically dominant subset of the virtual processor's instructions execute on the host processor in native mode.

Thus, a VCS provides an efficient operation of one or more copies of a complete computer system, similar to the host (or



Virtualization Overview: KVM Today

Native vs. Virtualized: 1% Overhead in KVM



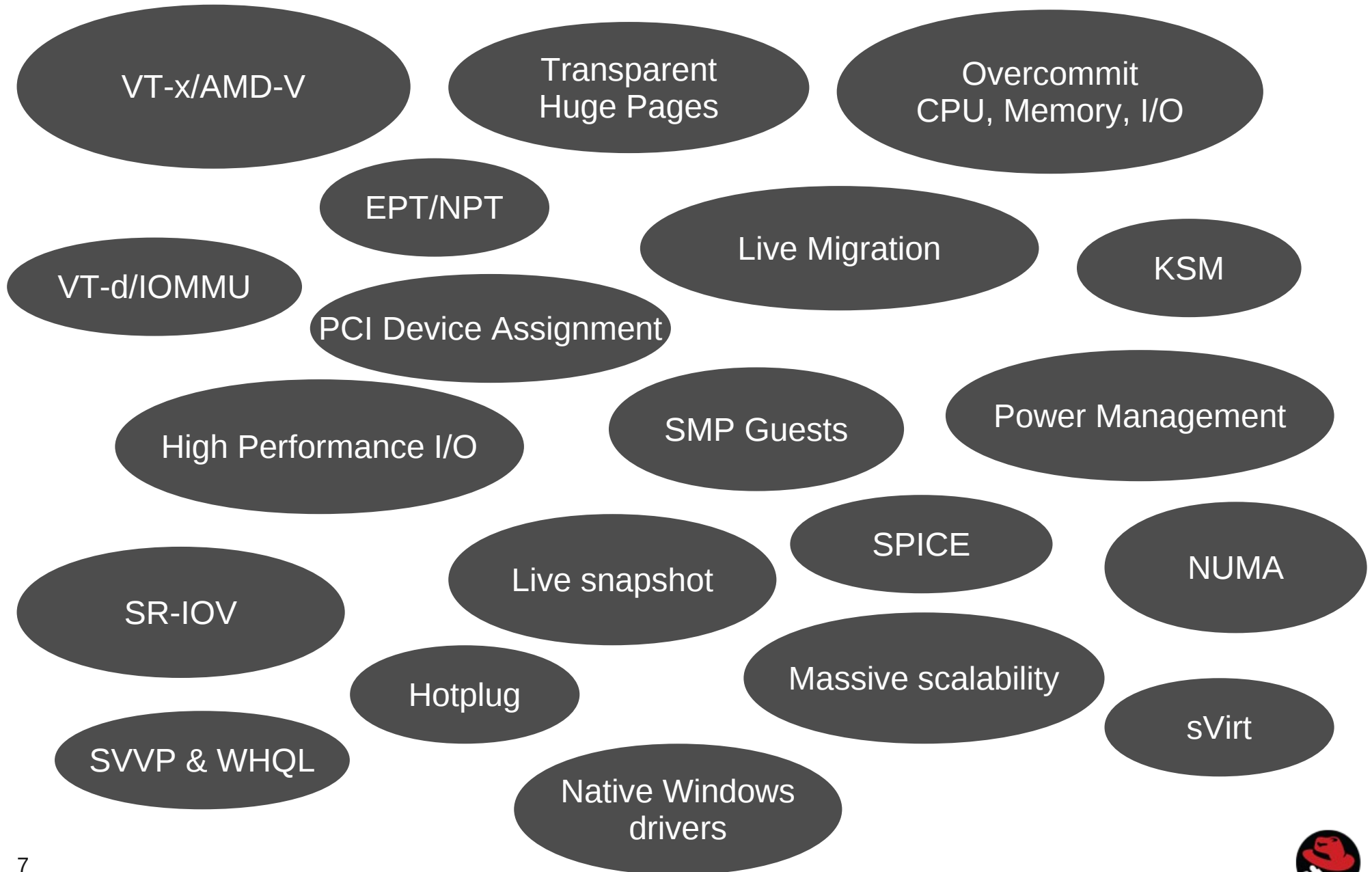
SAP Linux Certification Suite
CPU intensive reports phase
RHEL 6 + KVM and Intel E7-8800



KVM: The Kernel-based Virtual Machine



KVM features

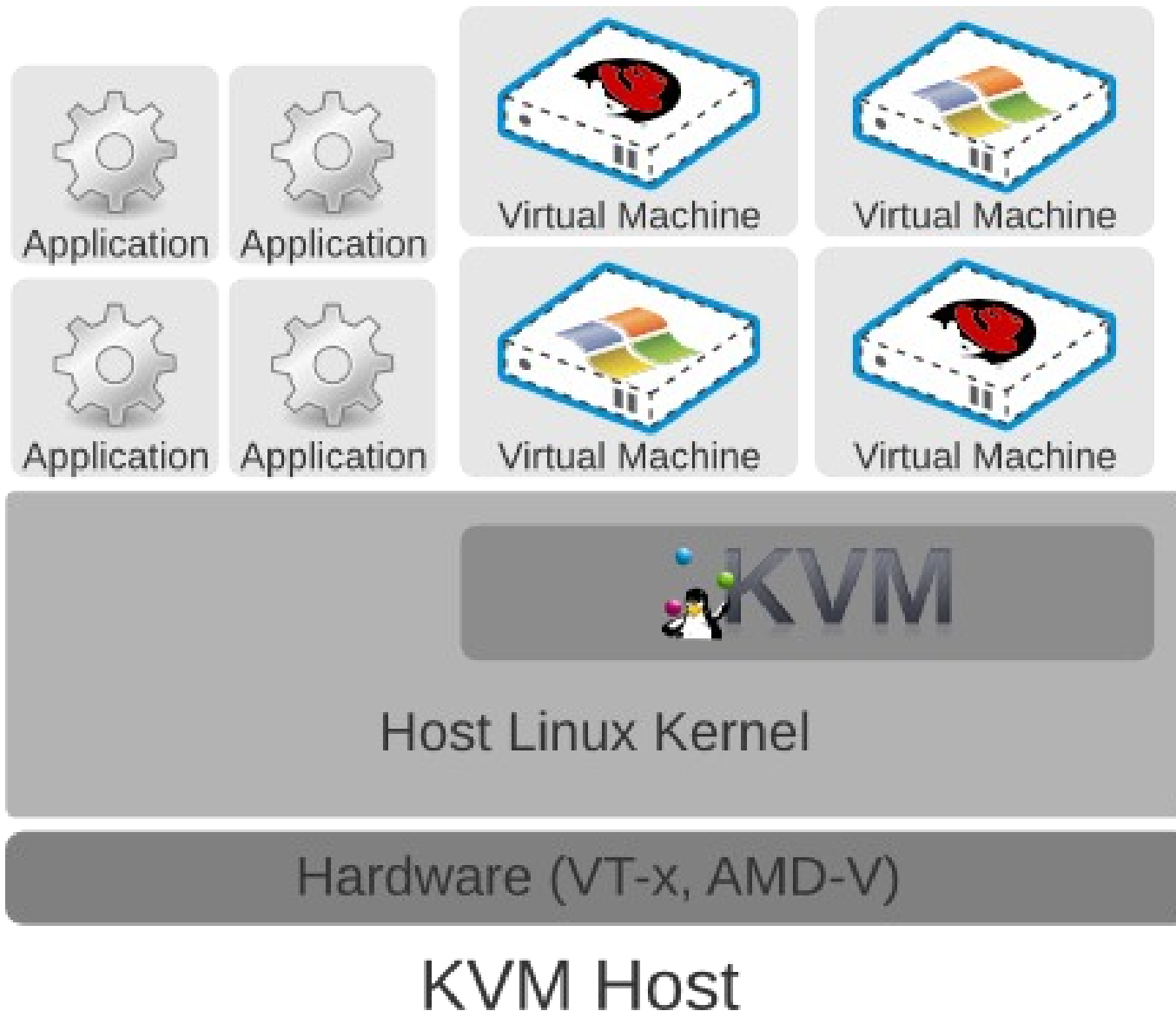


Why **reinvent** the wheel?

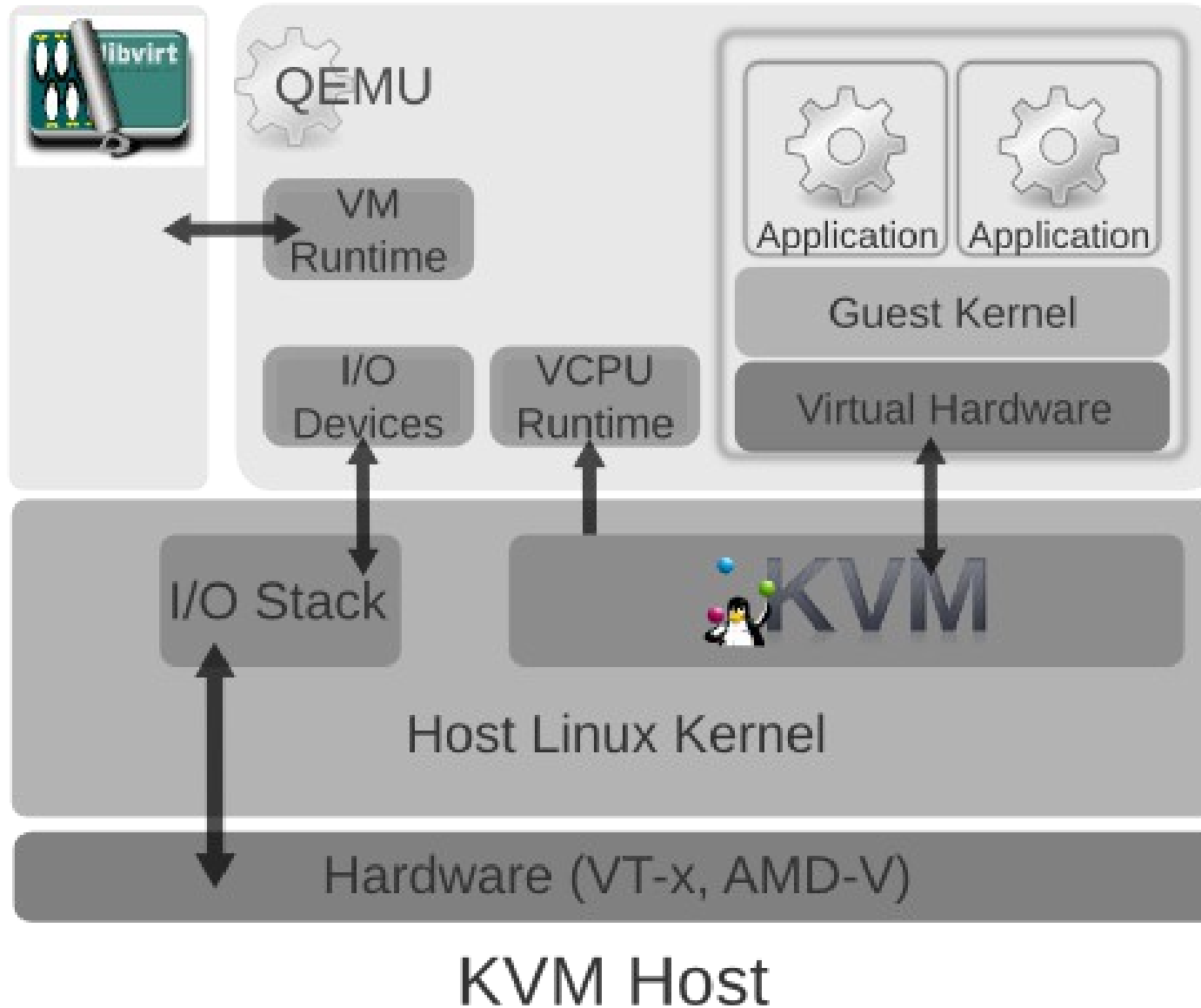
Focus on virtualization.



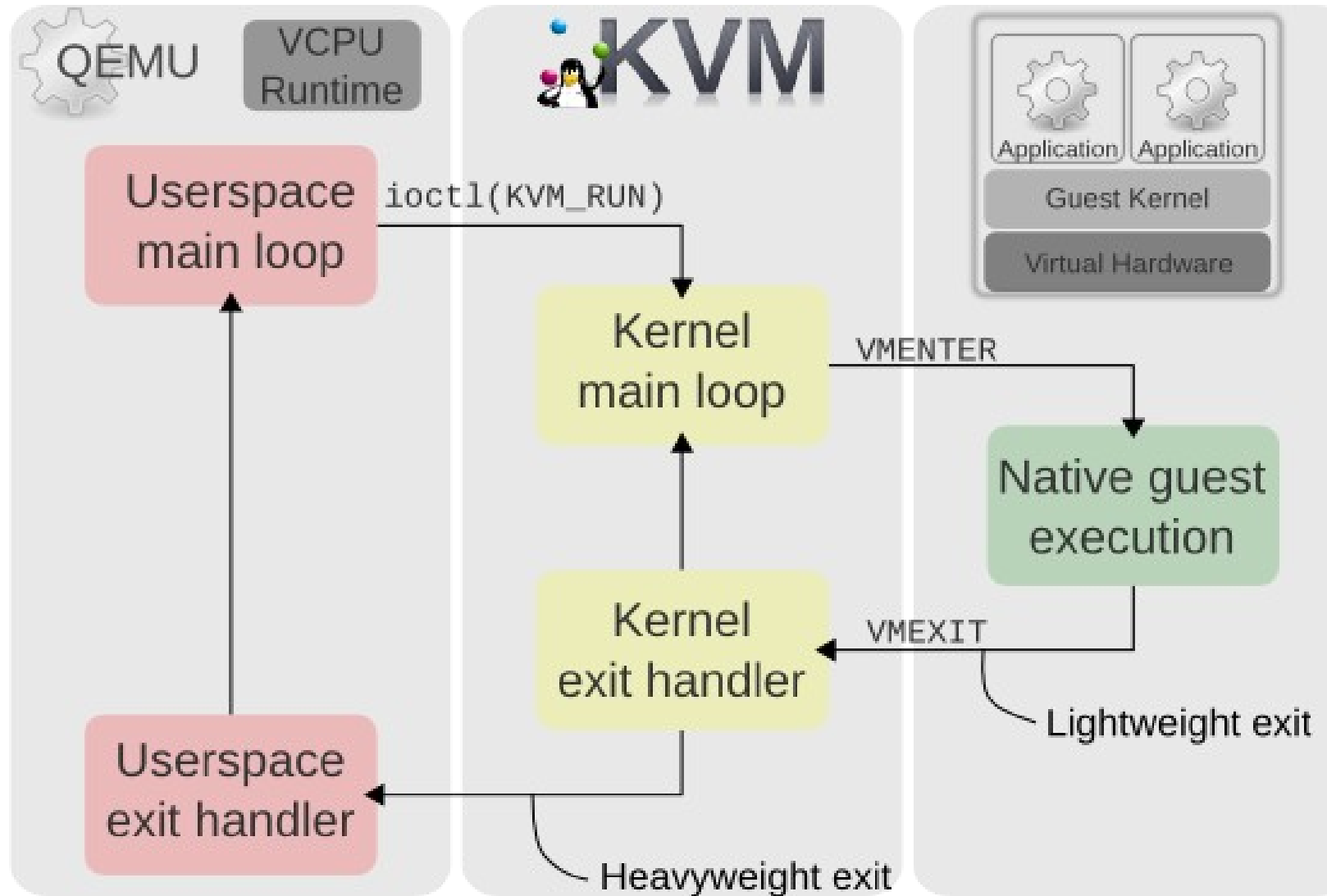
KVM Architecture



KVM Architecture



KVM Architecture



KVM in the Real World

- RAS
- Security Isolation
- Resource management
- Performance improvements
 - CPU, Block, Net, Memory
- SPECvirt



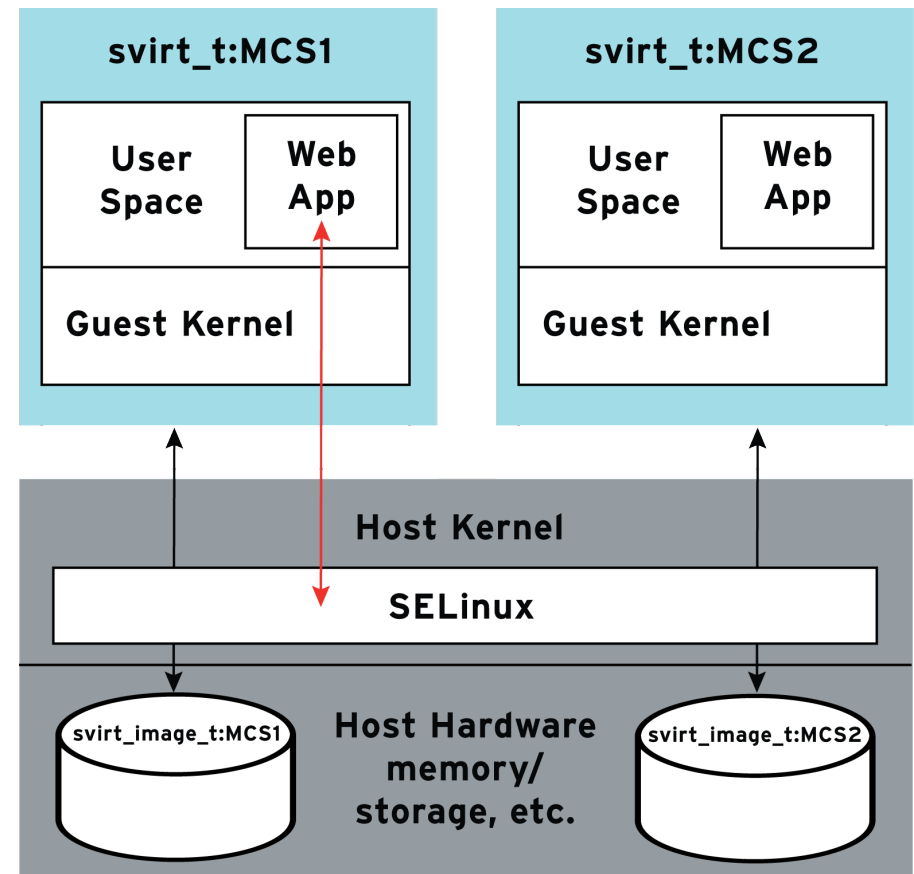
RAS

- timekeeping
- QMP, qdev, VMState, SeaBIOS
- vmchannel
- stable guest HW ABI
 - machine type
 - stable PCI topology
- PCI Device Assignment improvements
- Live migration with large memory guests



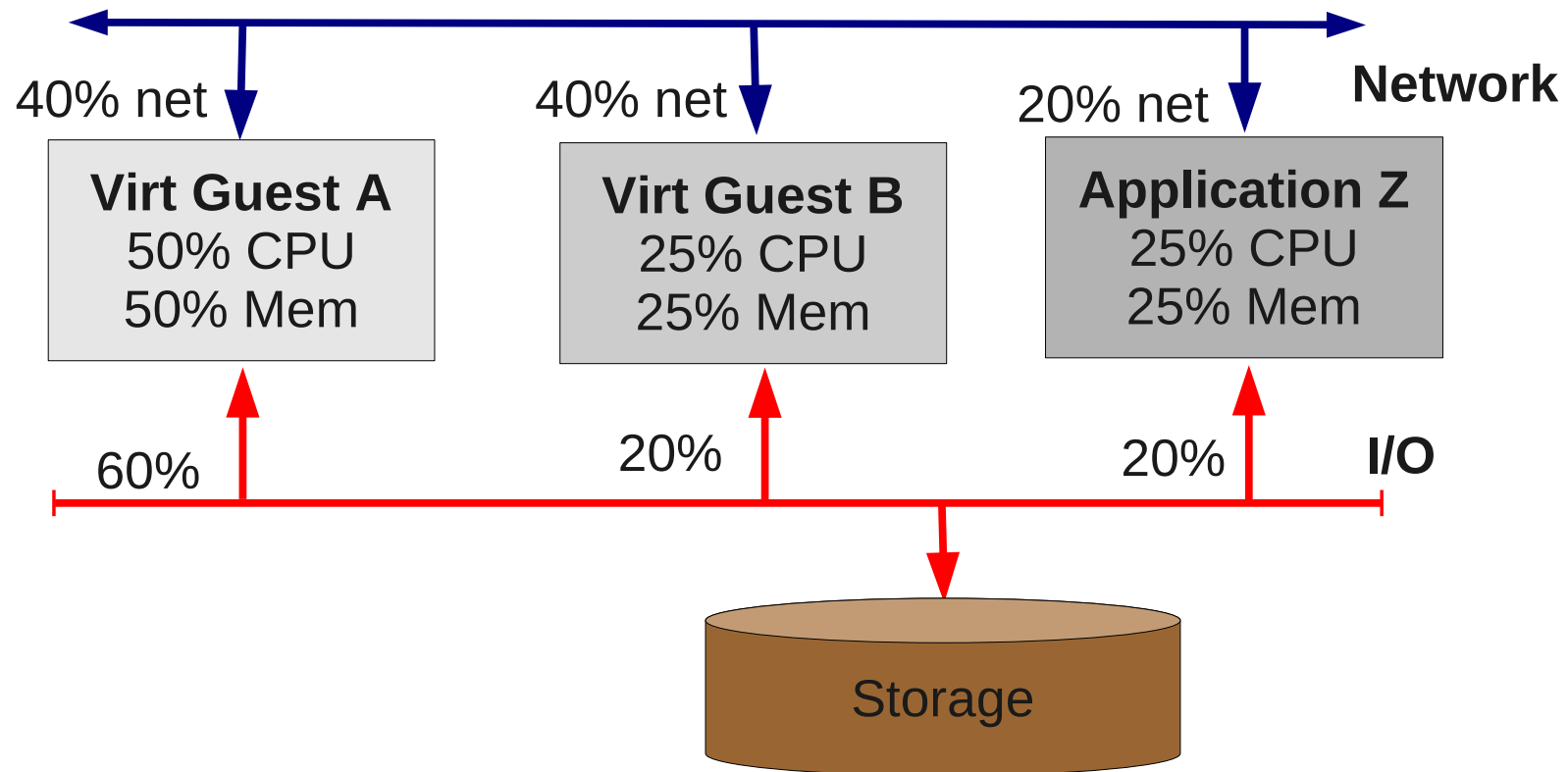
Isolation: sVirt Security (or similar w/ AppArmour)

- *Applying security labels to individual guest virtual machines and their resources*
 - Guest Isolation achieved with SELinux Mandatory Access Controls (MAC)
 - Protect against untrusted Guest VM
 - Protect against Host misconfiguration
 - Prevents unauthorized access of Guests/Host
 - Builds on existing, proven security mechanisms & controls



Isolation: Resource Management

- Control Group (Cgroups) for CPU/Memory/Network/Disk
 - Benefit: guarantee Quality of Service
 - Ideal for: Virtualization/Cloud deployments



KVM: CPU Performance

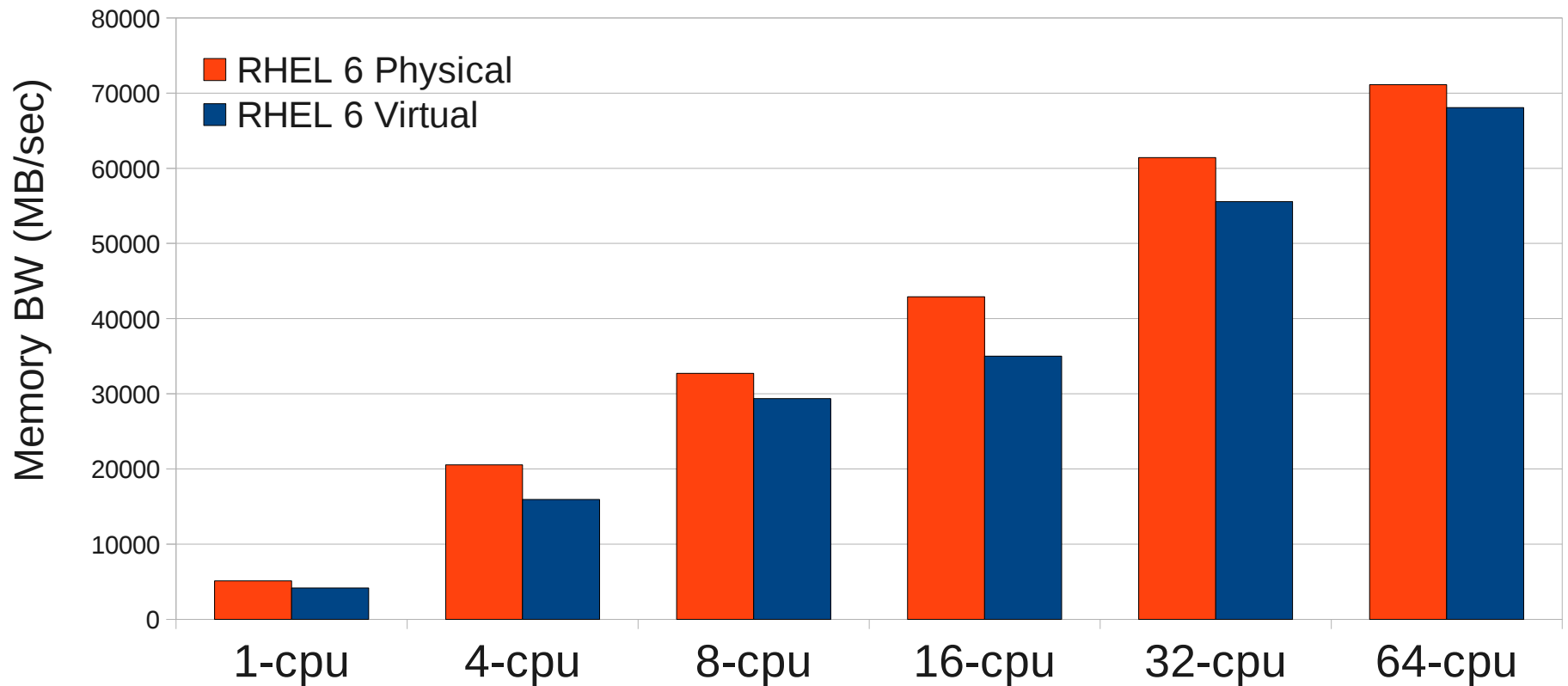
- Scale to 64 vcpus! And same pcpu scaling as Linux
- Guest spin lock-holder preemption sol'n
- KVM efficiency
 - User return notifiers...get lazier
- x2apic
 - Use MSR access to limit mmio accesses to the irq chip



Performance: SMP Scalability

64 CPU Scalability - Stream Benchmark

Intel EX 64-cpu, 128GB, FC

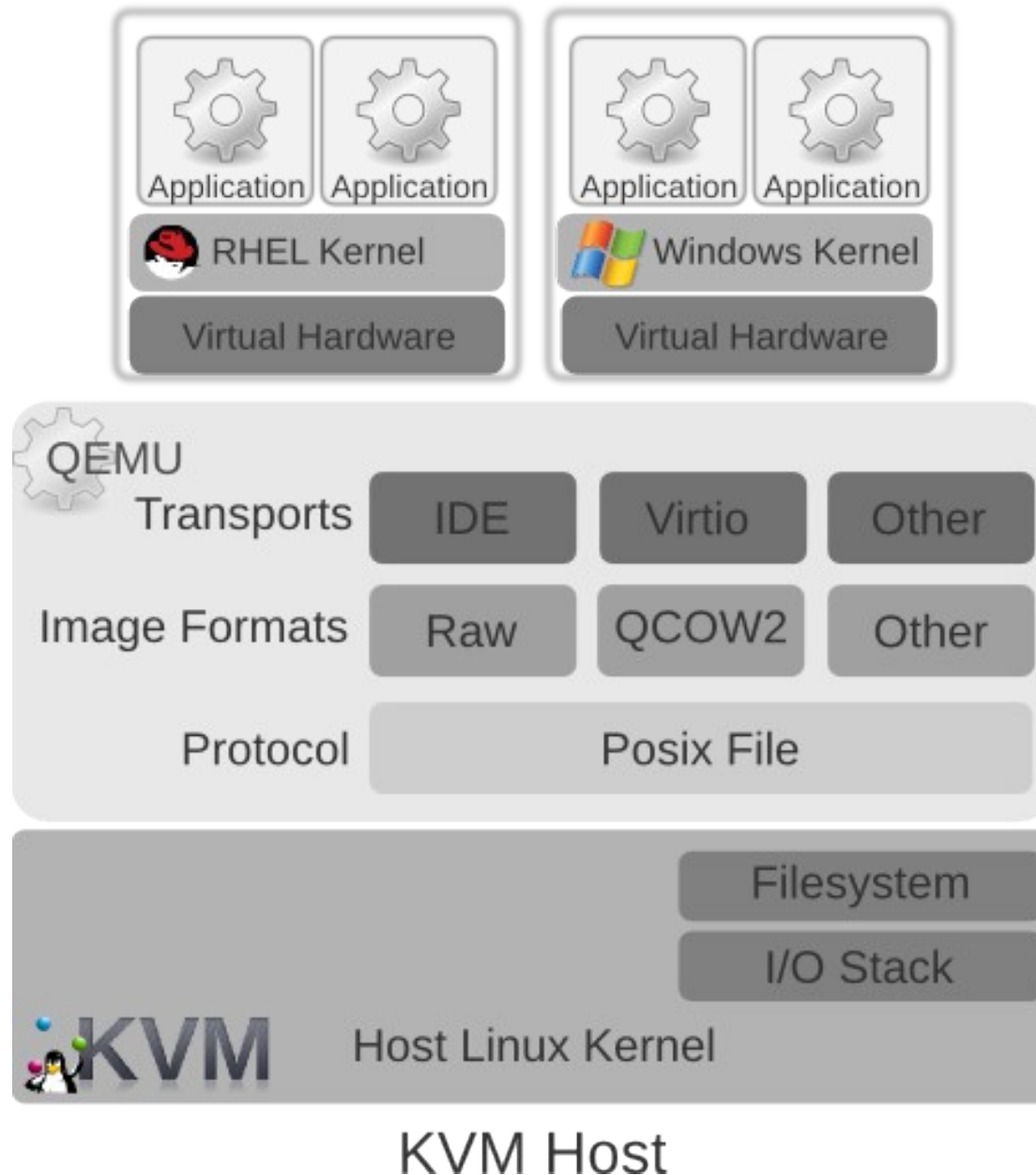


Excellent, linear scalability; minimal virtualization overhead

Note: With this h/w, at 32 CPUs socket bandwidth is saturated



KVM Block Architecture



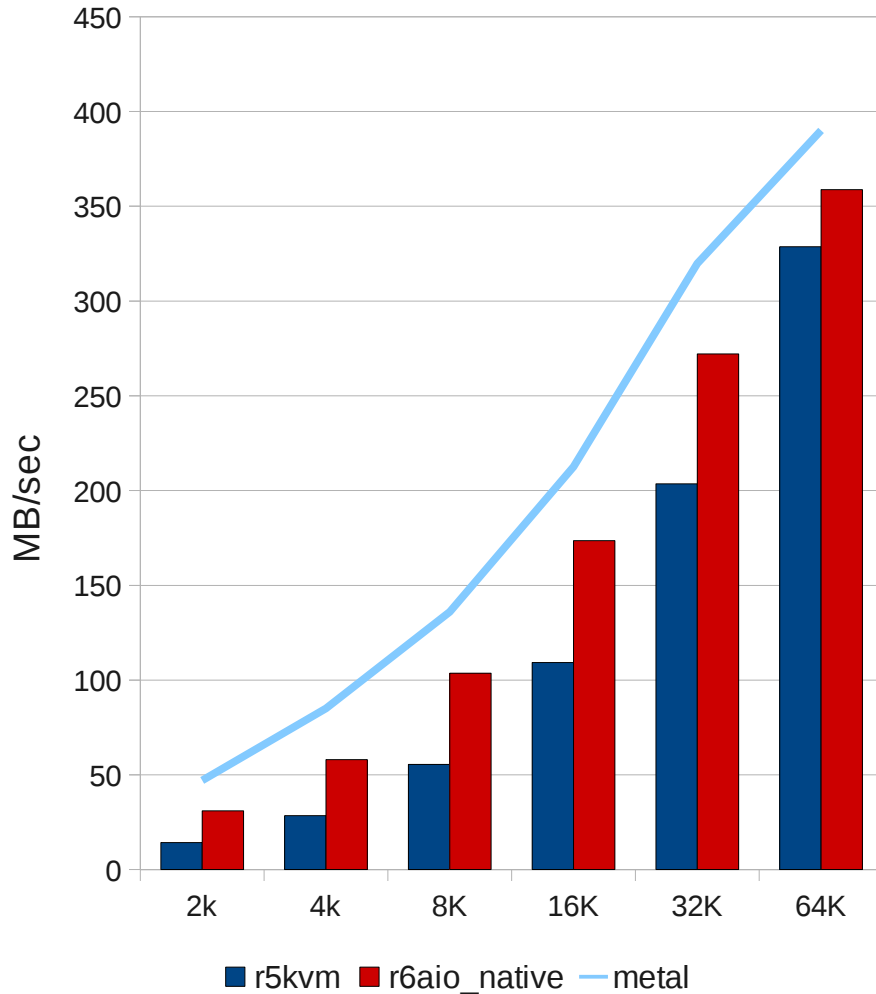
KVM: Block Performance

- qemu vectored I/O (preadv/pwritev) and native AIO
- virtio
 - flush + fua
 - MSI support
 - ioeventfd
- qcow2 cache

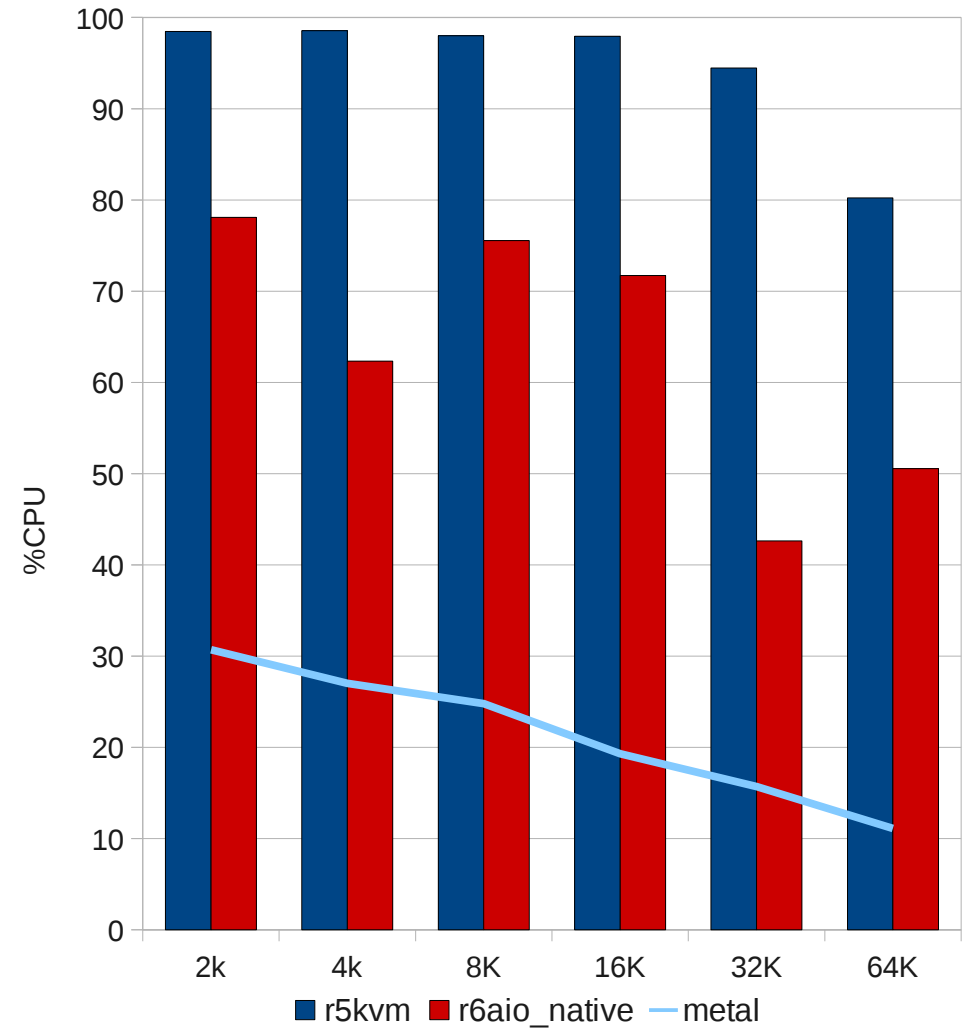


KVM Performance – RHEL6 aio=native Win2k8 Intel 24cpu, 64GB, FC IOmeter

IO - Sequential Reads

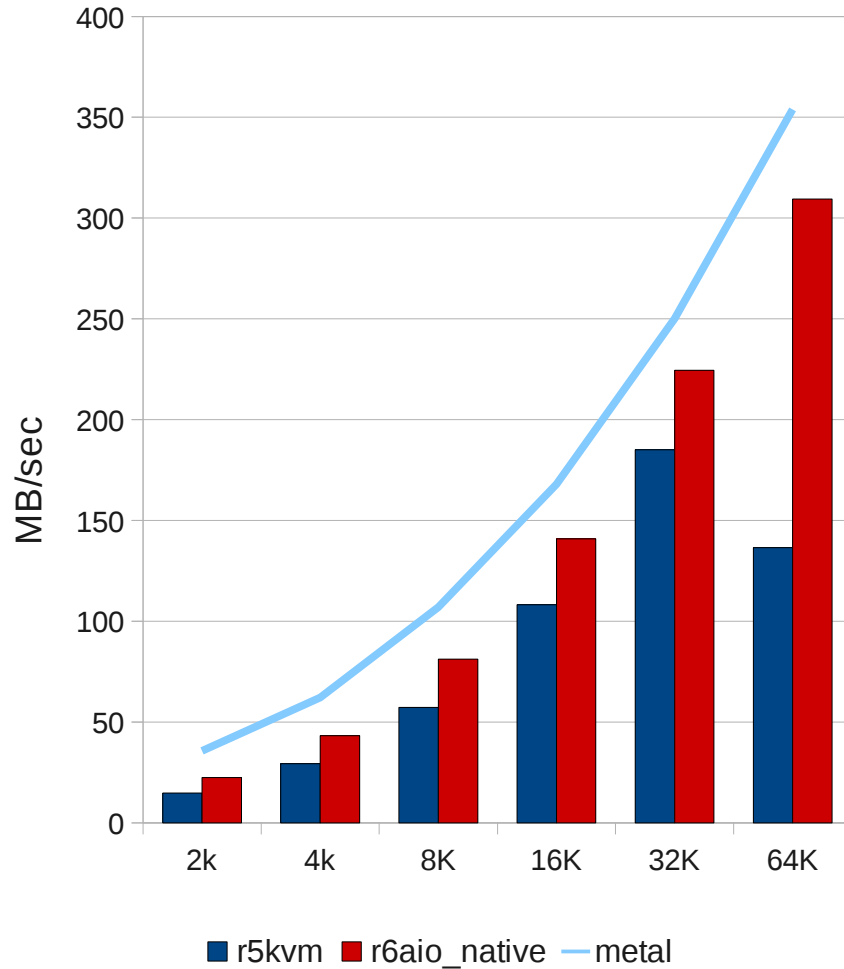


CPU - Sequential Reads

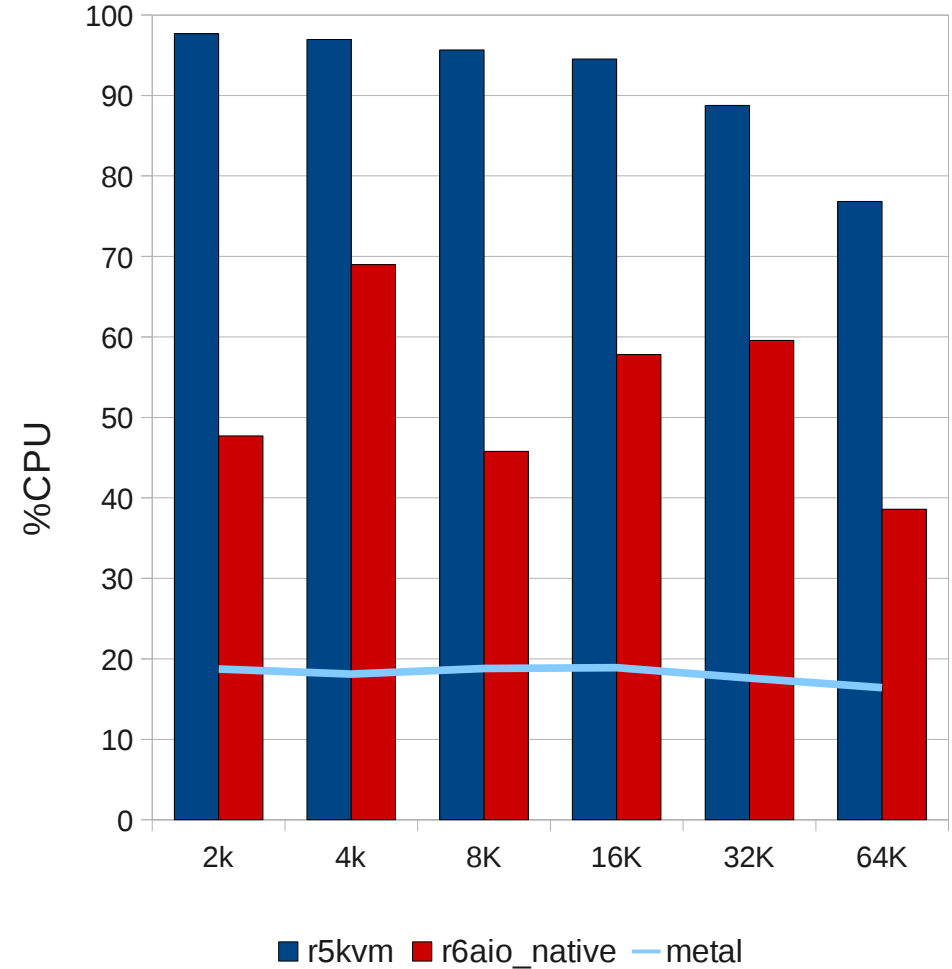


KVM Performance – RHEL6 aio=native Win2k8 Intel 24cpu, 64GB, FC IOmeter

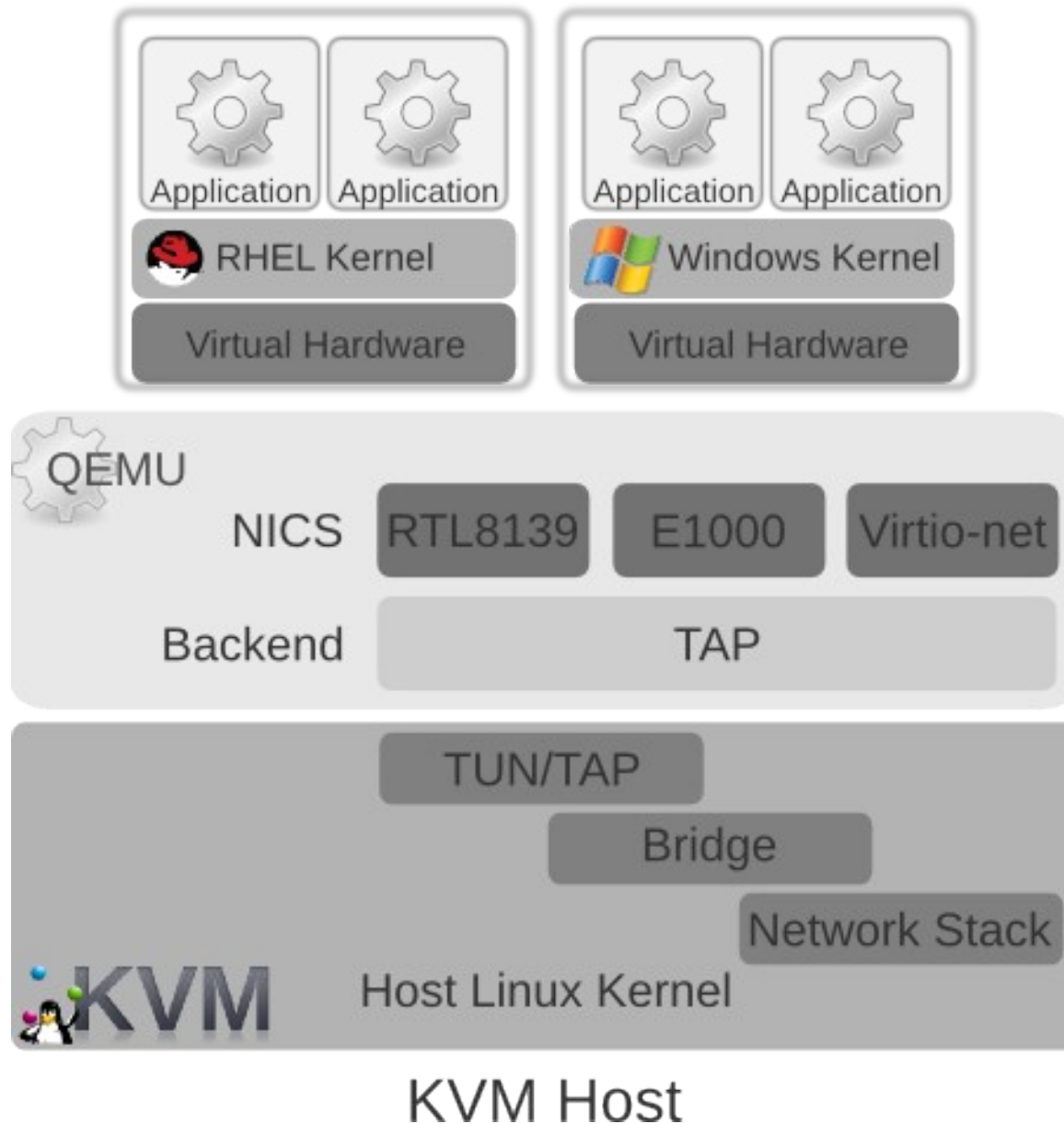
IO - Sequential Writes



CPU - Sequential Writes



KVM Network Architecture

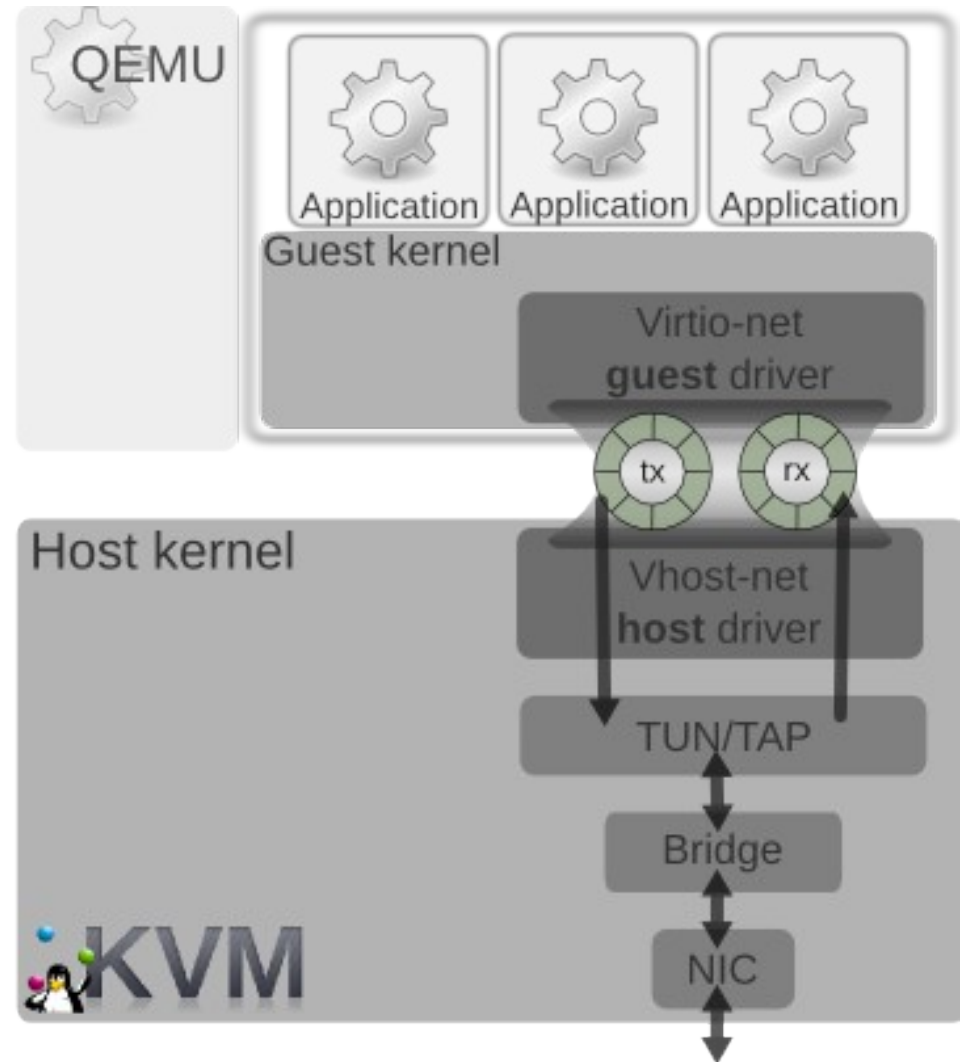
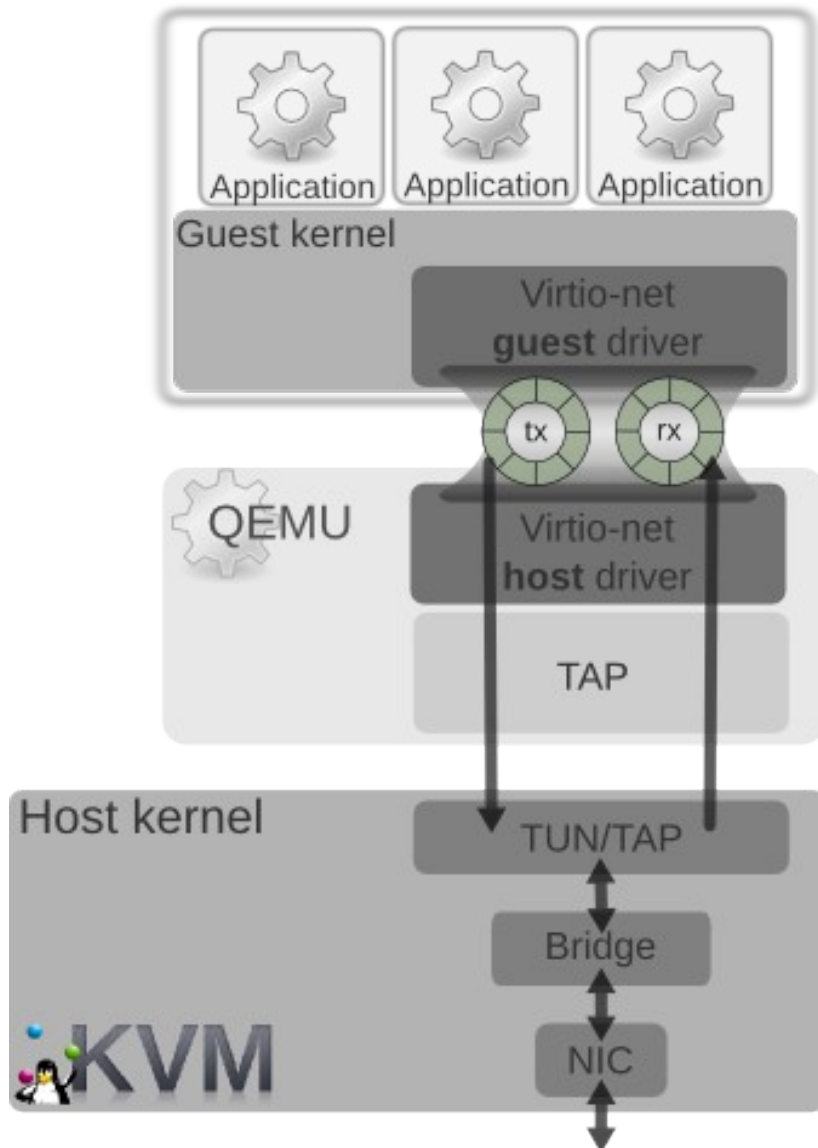


Performance: Networking

- vhost-net
 - virtio backend in kernel
 - transparent to guest
 - reduce latency
 - better throughput
 - lower CPU consumption
- SR-IOV for near native performance
- VEPA/VNLink



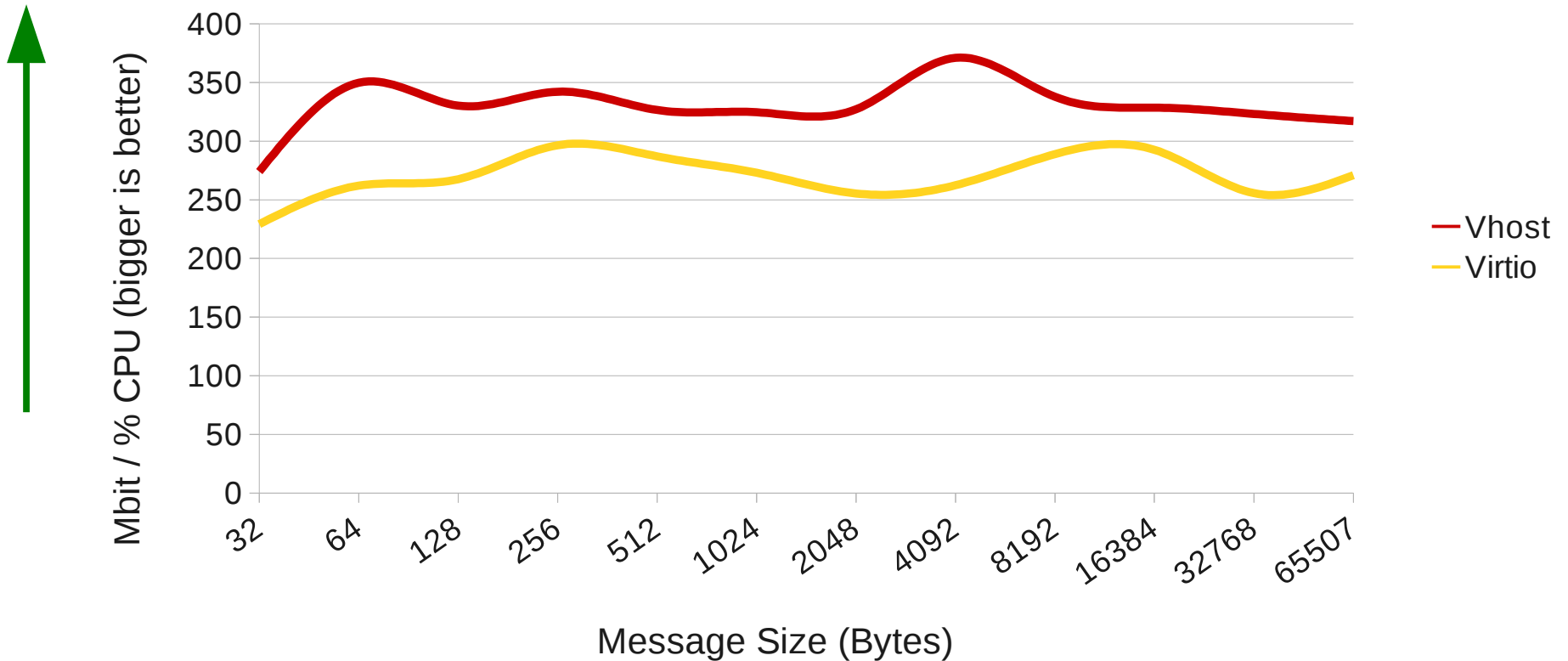
virtio network architecture – detail



vhost_net Efficiency

8 Guest Scale Out RX Vhost vs Virtio - % Host CPU

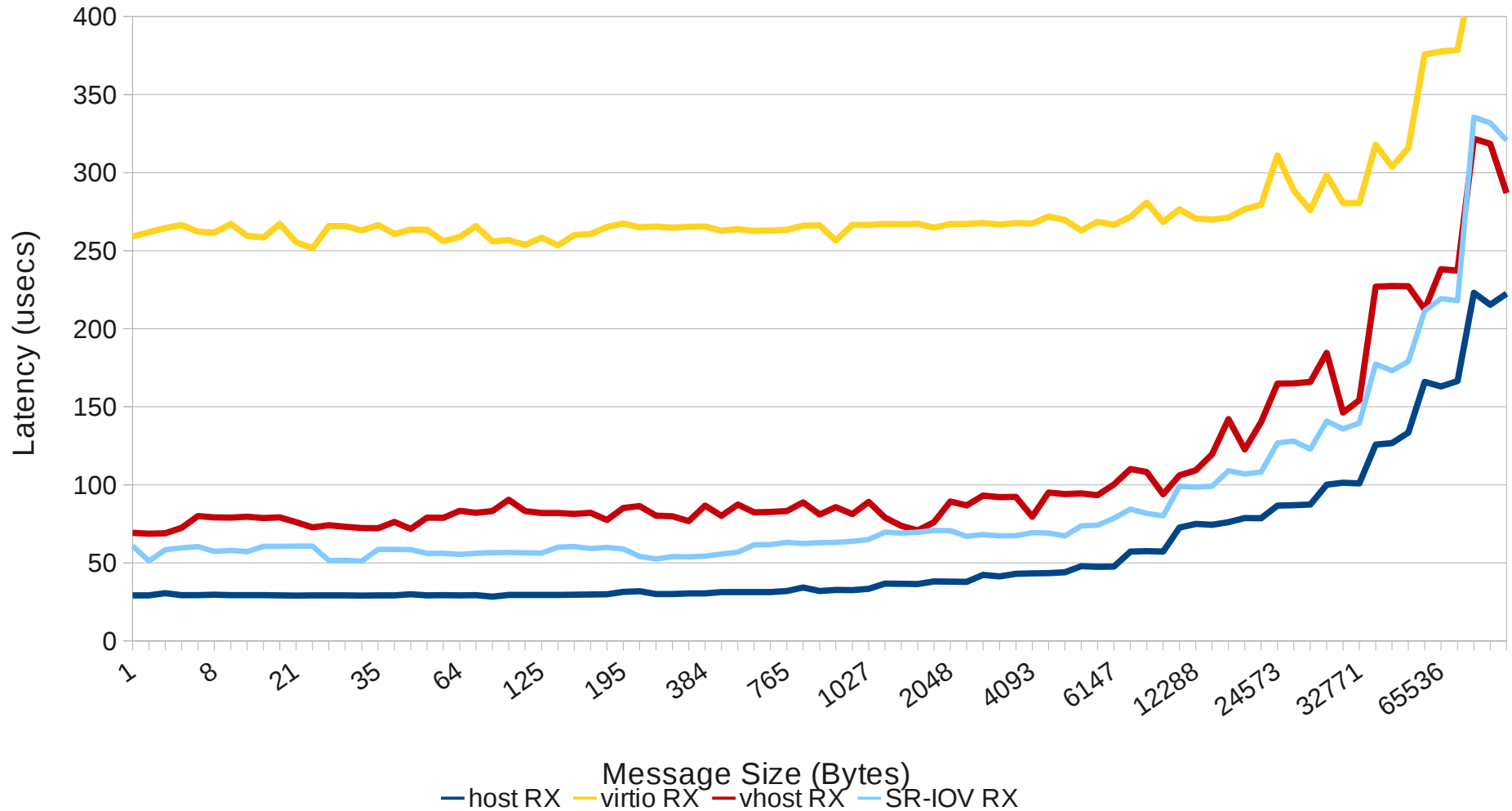
Mbit per % CPU netperf TCP_STREAM



Latency comparison – RHEL 6

Network Latency by guest interface method

Guest Receive (Lower is better)



Performance: Memory

- Transparent Huge Pages
 - Benefit for bare metal apps (4.2%)
 - Huge benefit for KVM VMs (25%)
- KSM working with THP



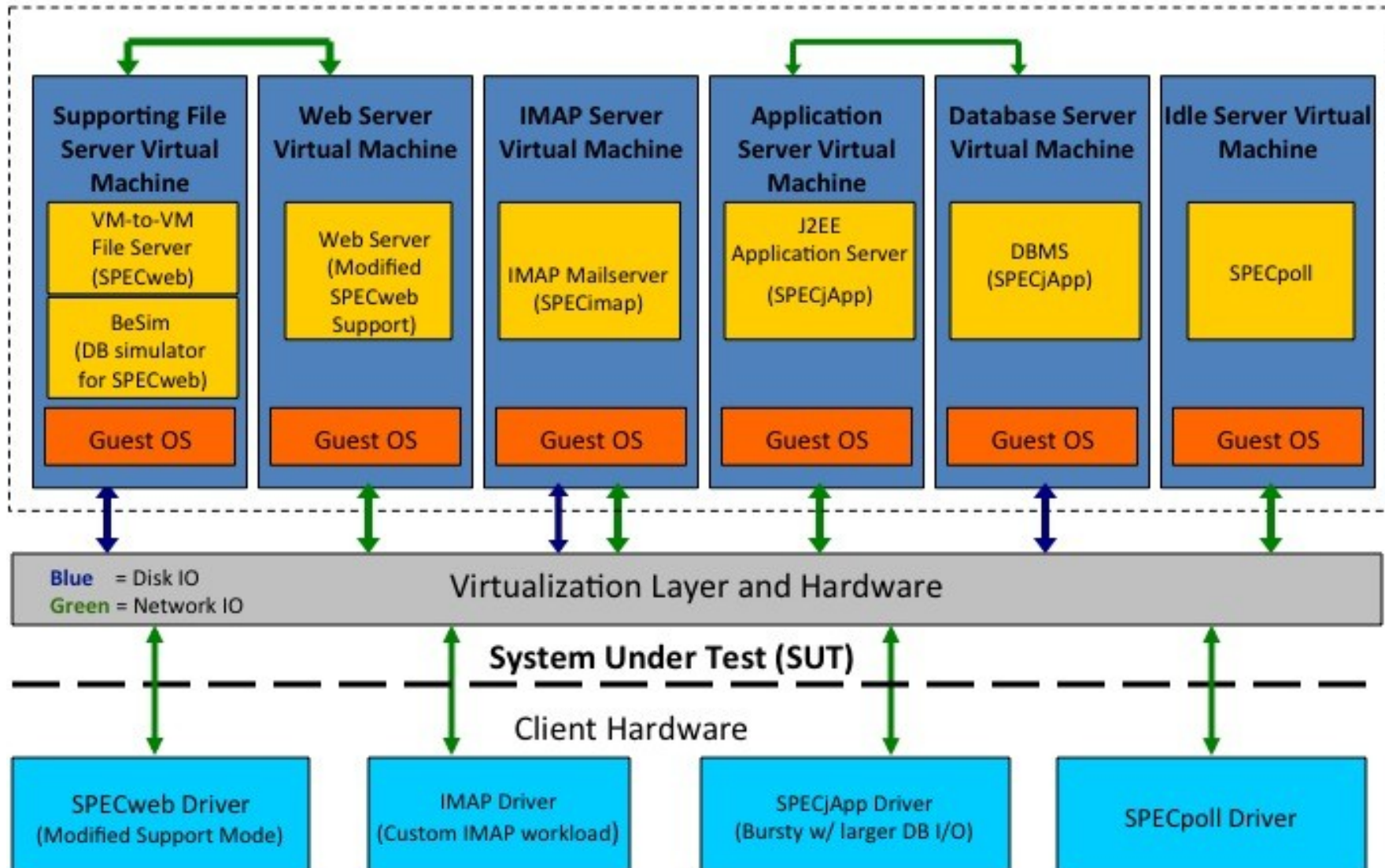
Performance

What does it **add** up to?

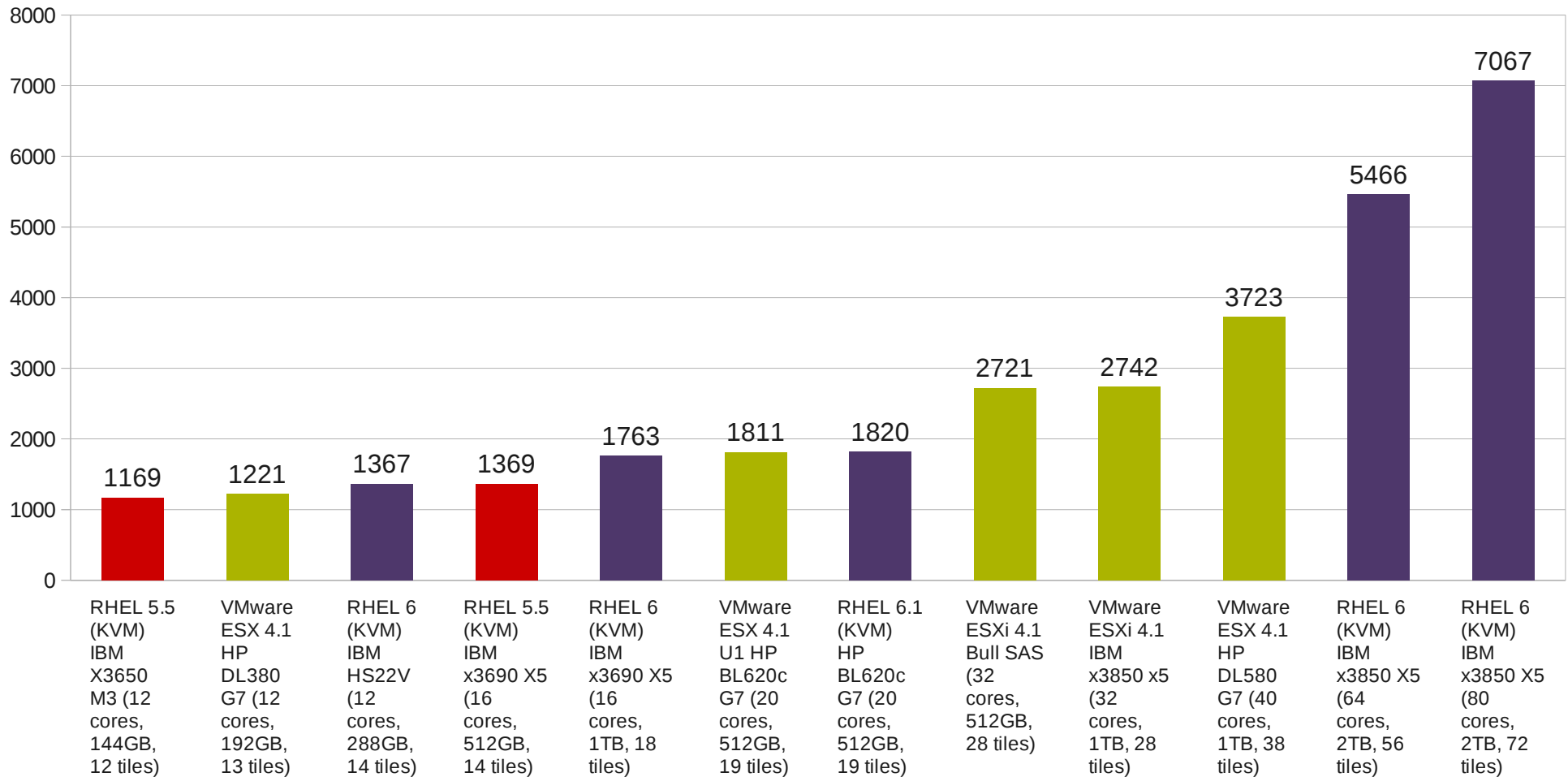


SPECvirt

SPECvirt_sc2010 TILE



SPECvirt_sc2010 results*



*As of May 4, 2011. See http://www.spec.org/virt_sc2010 for details



Agenda

Virtualization Overview

Foundation for Cloud

Future



Cloud

- Computing as a Service (Anything as a Service)
- Utility model
- Self provisioning
- Dynamic resources



KVM in Cloud

- Increase density
- Improve efficiency
- Multi-tenancy (security, isolation)



KVM Is The Cloud



Agenda

Virtualization Overview

Foundation for Cloud

Future



Roadmap

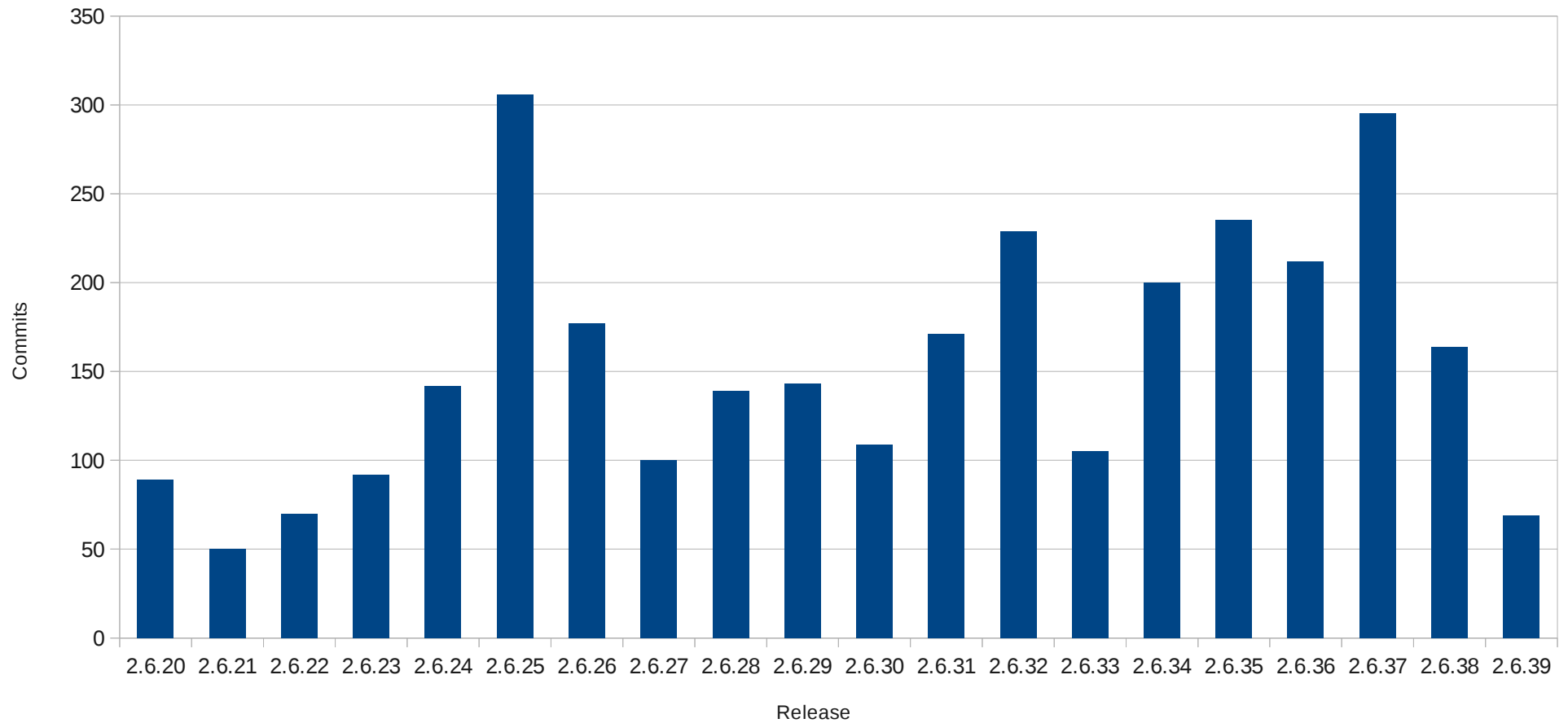
- Scaling and performance
- NUMA
- Lock holder preemption
- Zero copy networking
- Multiqueue virtio-net
- Debugging and profiling
- Further cgroup integration
- virt-agent (guest agents)
- QCOW2 performance
- Multi-tenant network isolation
- Memory management (async page faults, hinting, compcache)
- USB 2.0 support
- Performance Monitoring Unit (PMU) support

Weather forecast...



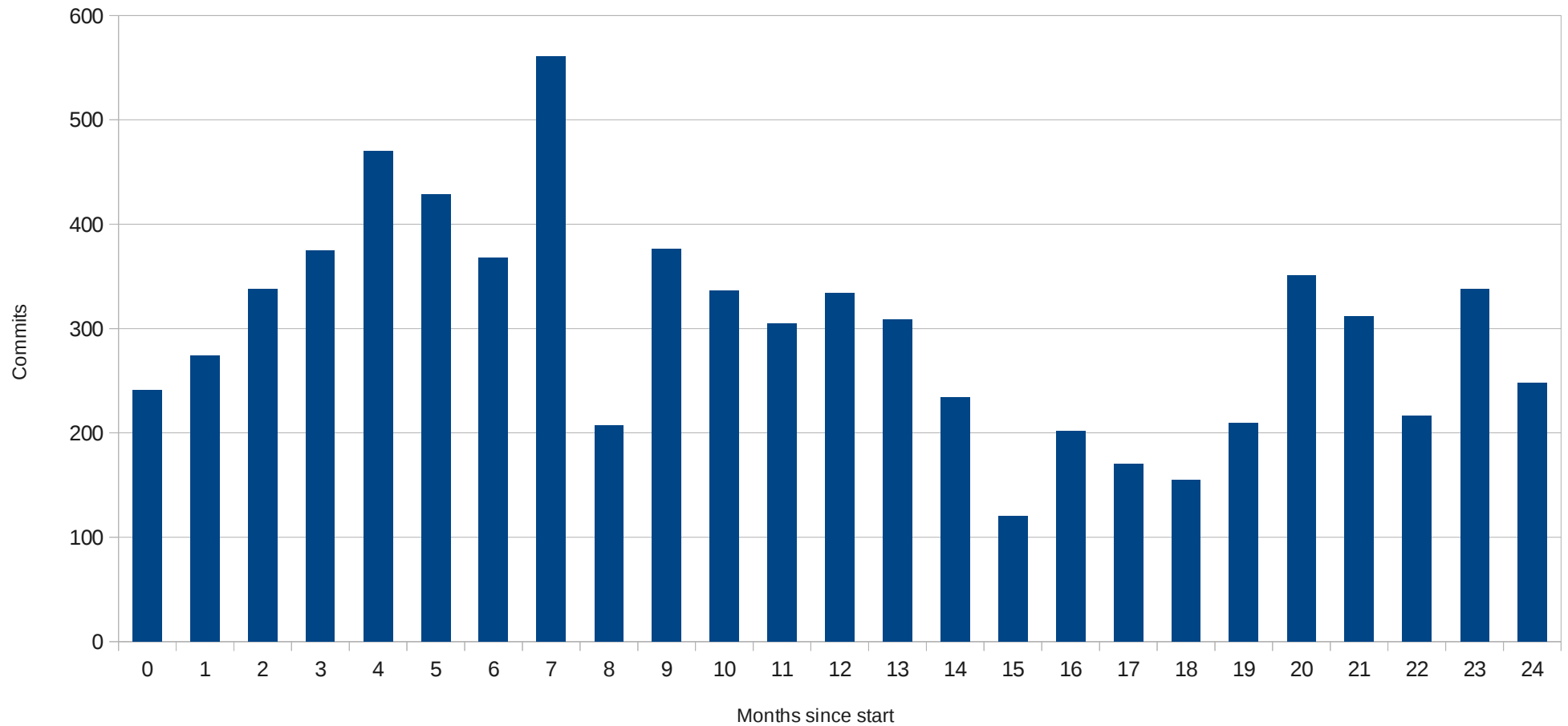
KVM development activity

KVM commits per release



QEMU development activity

QEMU commits per month



Cloudy
with a chance of
total world domination

