

KVM概述



概述

- ▶ 虚拟化概述
- ▶ KVM概述
- ▶ 资源与参考资料

◆ 虚拟化概述

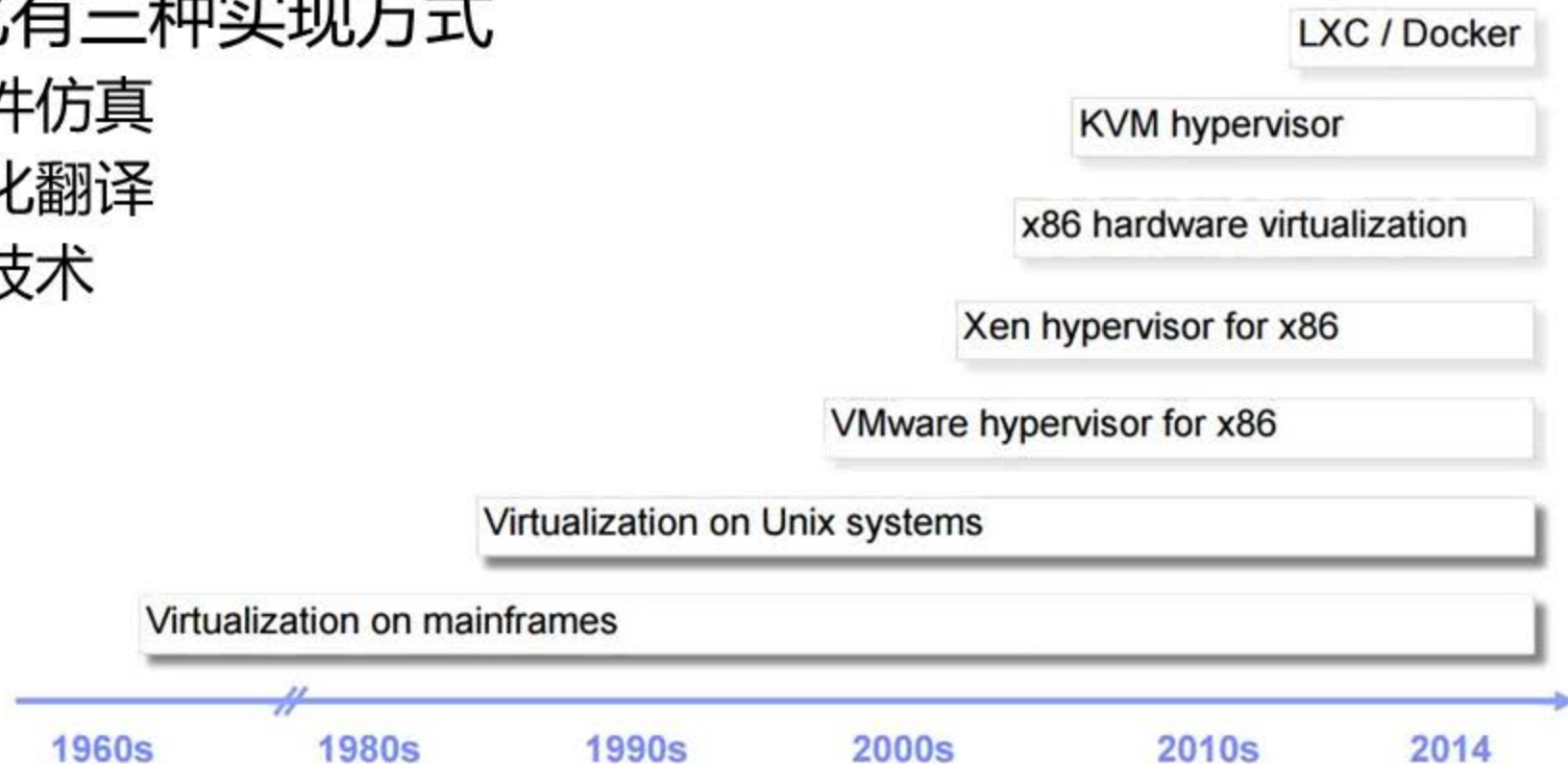
- ▶ 什么是虚拟化
- ▶ 虚拟化技术历史
- ▶ 虚拟化实现方式

虚拟化的定义

- ▶ 在计算技术中，虚拟化意味着创建设备或资源的虚拟版本，如服务器、存储设备、网络或者操作系统等等
- ▶ 虚拟化技术
 - ▶ 系统虚拟化
 - ▶ 存储虚拟化
 - ▶ 网络虚拟化
 - ▶ GPU虚拟化
 - ▶ 软件虚拟化
 - ▶ 硬件支持虚拟化

系统虚拟化

- ▶ 这种虚拟化通常表现为在单一系统上运行多个操作系统
- ▶ 这些虚拟操作系统同时运行，每个操作系统又是相互独立
- ▶ 虚拟化有三种实现方式
 - ▶ 纯软件仿真
 - ▶ 虚拟化翻译
 - ▶ 容器技术



纯软件仿真

- ▶ 通过模拟完整的硬件环境来虚拟化来宾平台。
- ▶ 模拟X86、ARM、PowerPC等多种CPU
- ▶ 效率比较低
- ▶ 产品或方案
 - ▶ QEMU、Bochs、PearPC

For system emulation, the following hardware targets are supported:

- PC (x86 or x86_64 processor)
- ISA PC (old style PC without PCI bus)
- PREP (PowerPC processor)
- G3 BW PowerMac (PowerPC processor)
- Mac99 PowerMac (PowerPC processor, in progress)
- Sun4m/Sun4c/Sun4d (32-bit Sparc processor)
- Sun4u (64-bit Sparc processor, in progress)
- Malta board (32-bit and 64-bit MIPS processors)
- ARM Integrator/CP (ARM)
- ARM Versatile baseboard (ARM)
- ARM RealView Emulation baseboard (ARM)
- Spitz, Akita, Borzoi and Terrier PDAs (PXA270 processor)
- Luminary Micro LM3S811EVB (ARM Cortex-M3)
- Luminary Micro LM3S6965EVB (ARM Cortex-M3)
- Freescale MCF5208EVB (ColdFire V2).
- Arnewsh MCF5206 evaluation board (ColdFire V2).
- Palm Tungsten|E PDA (OMAP310 processor)

Downloads

RPM:

- **AIX5L V5.3:**
 - [gemu-0.9.1-1.aix5.3.ppc.rpm](#)
- **AIX V6.1:**
 - [gemu-0.9.1-1.aix6.1.ppc.rpm](#)
- **AIX V7.1:**
 - [gemu-0.9.1-1.aix7.1.ppc.rpm](#)

Source RPM:

- [gemu-0.9.1-1.src.rpm](#)

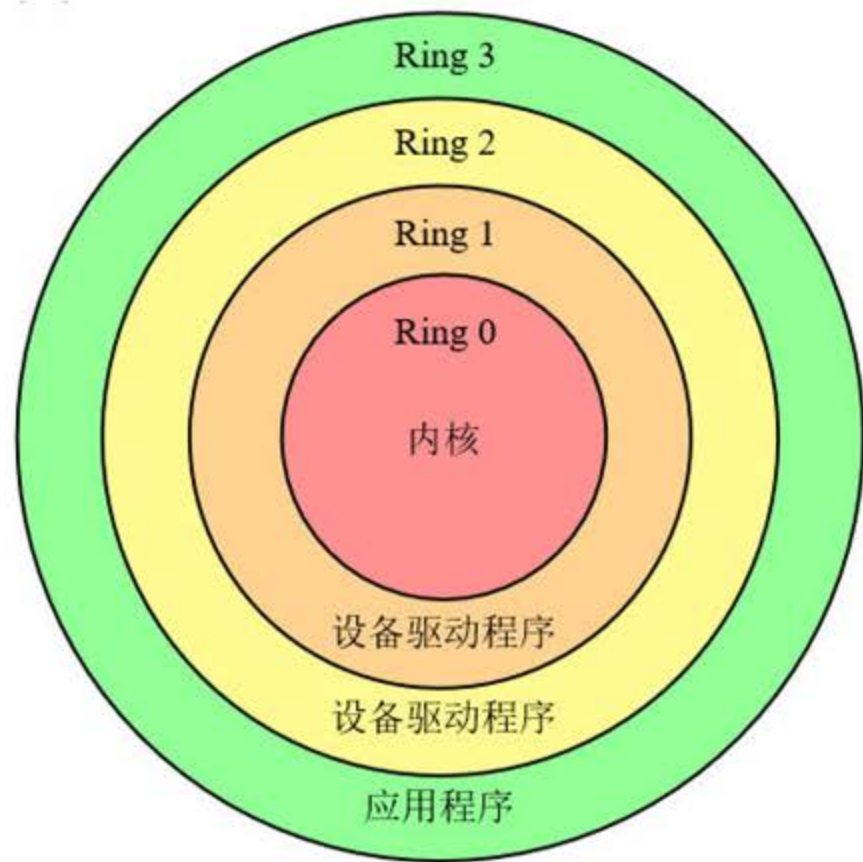
<http://www.perzl.org/aix/index.php?n=Main.Qemu>

虚拟化层翻译

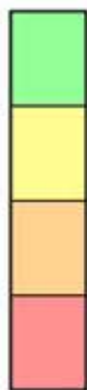
- ▶ 多数的虚拟化而采用虚拟机管理程序Hypervisor
- ▶ Hypervisor是一个软件层或子系统
 - ▶ 也称为VMM (Virtual Machine Monitor , 虚拟机监控器)
- ▶ 允许多种操作系统在相同的物理系统中运行
- ▶ 控制硬件并向来宾操作系统提供访问底层硬件的途径
- ▶ 向来宾操作系统提供虚拟化的硬件



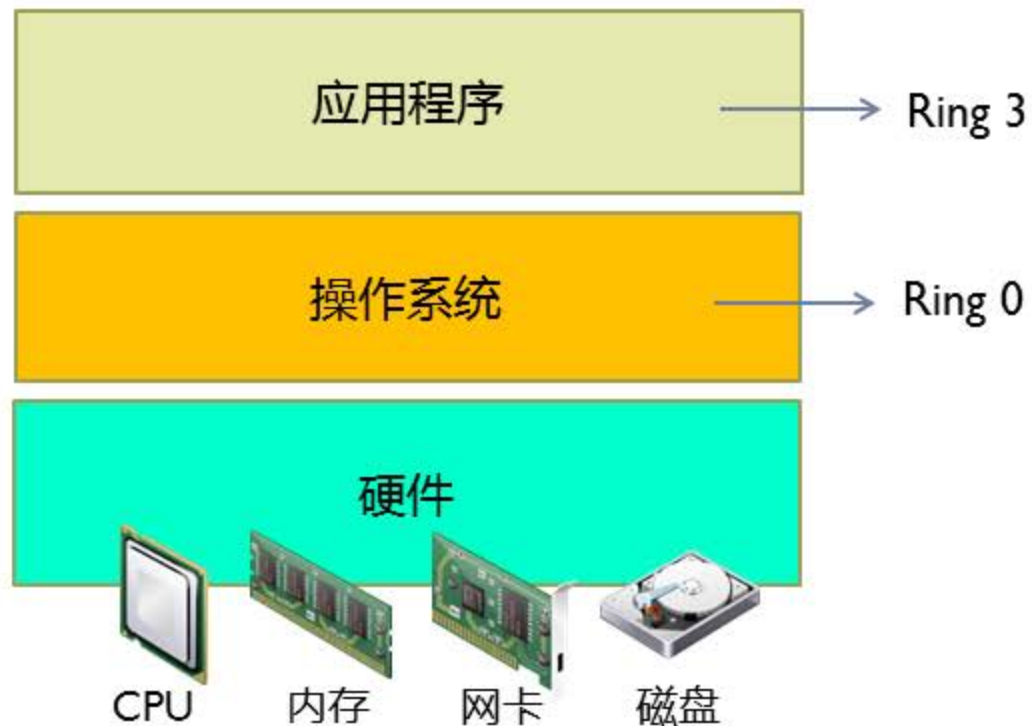
X86 CPU的保护环



最小权限

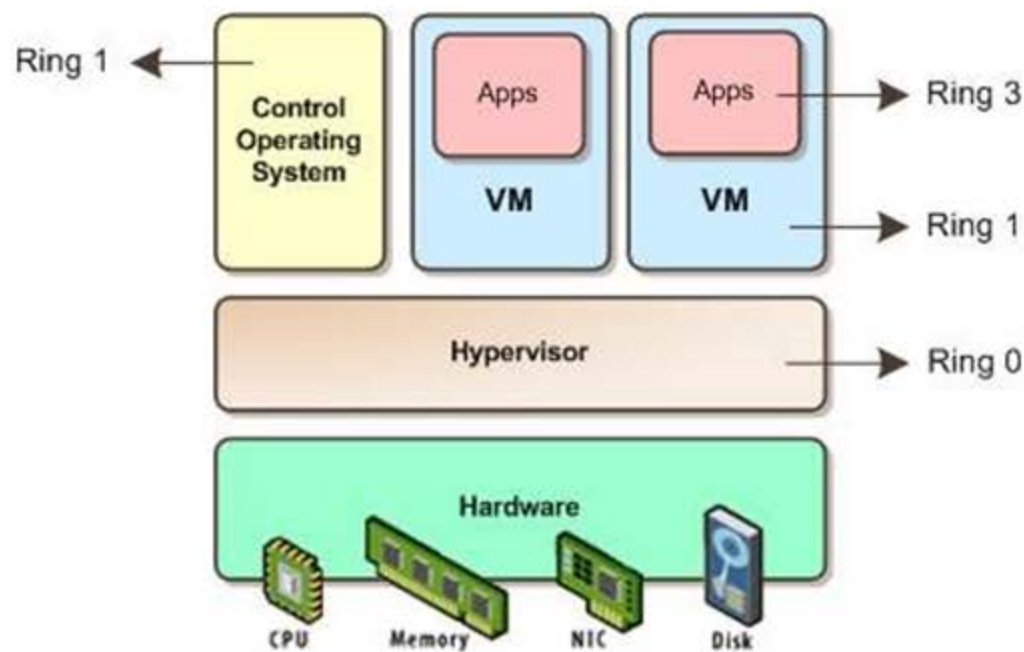


最大权限

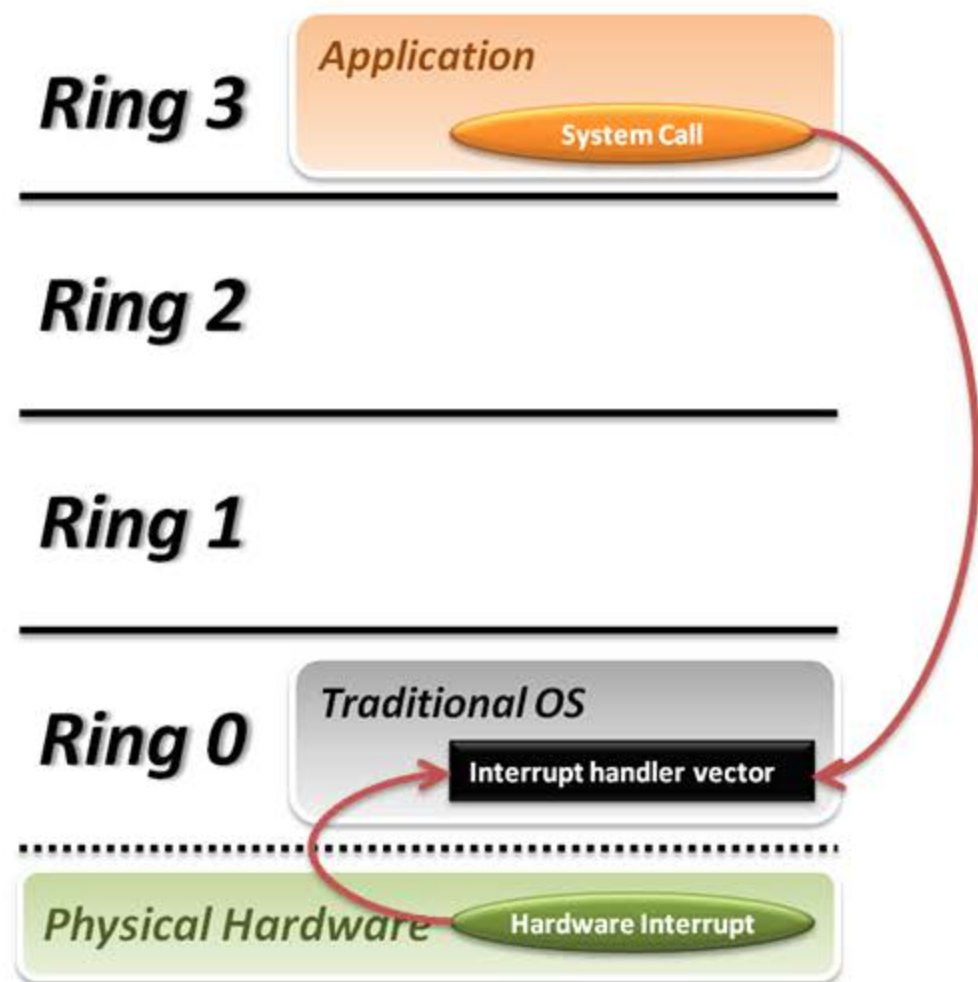


无硬件辅助的全虚拟化

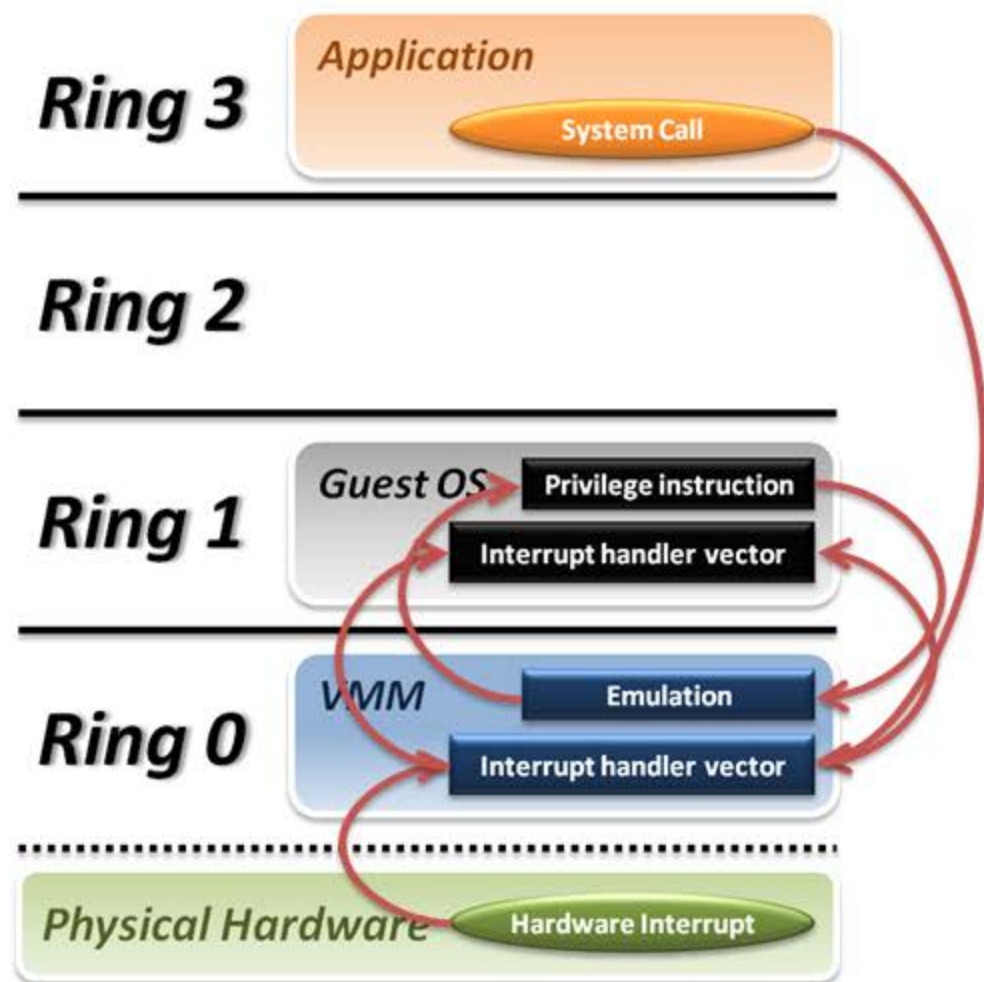
- ▶ Full Virtualization without Hardware Assist
- ▶ 基于二进制翻译的全虚拟化
(Full Virtualization with Binary Translation)
- ▶ Hypervisor运行在Ring 0
- ▶ Guest OS运行在Ring 1
- ▶ 机制：异常、捕获、翻译
- ▶ 示例：
 - ▶ VMware Workstation
 - ▶ QEMU
 - ▶ Virtual PC



Trap and Emulate模式



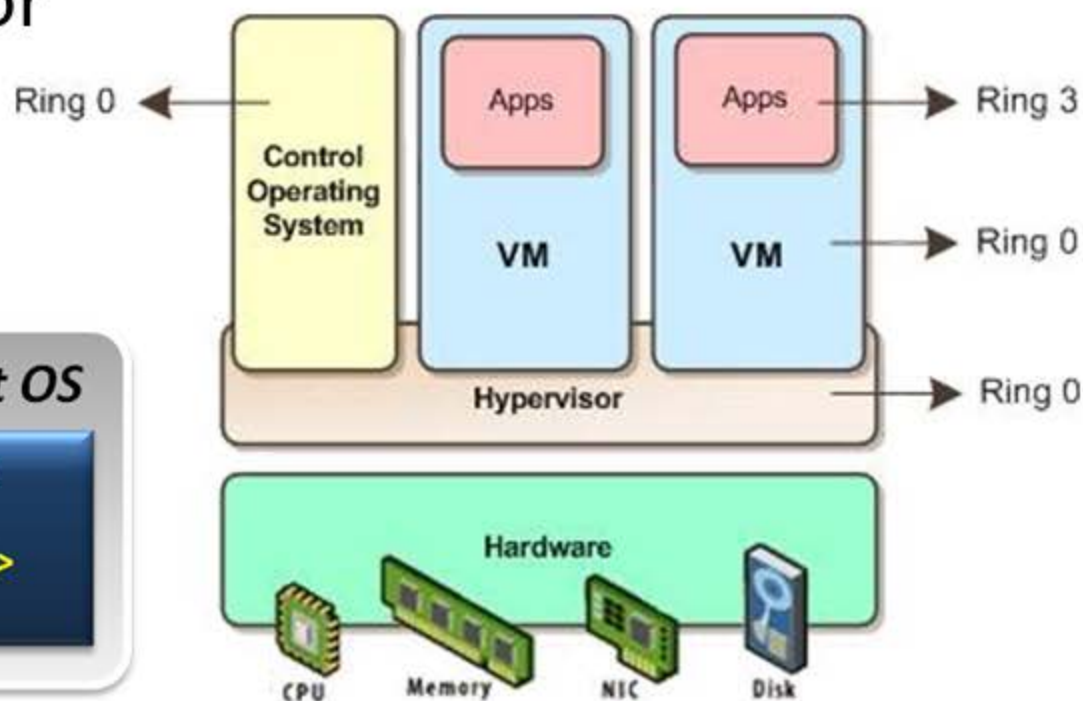
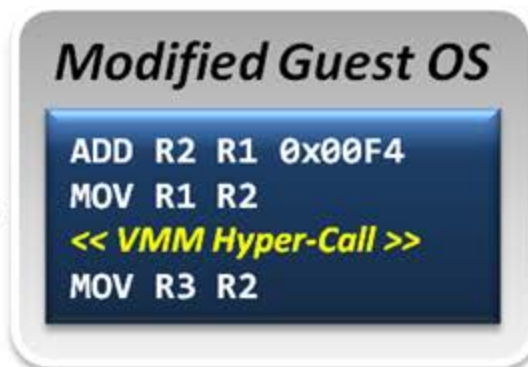
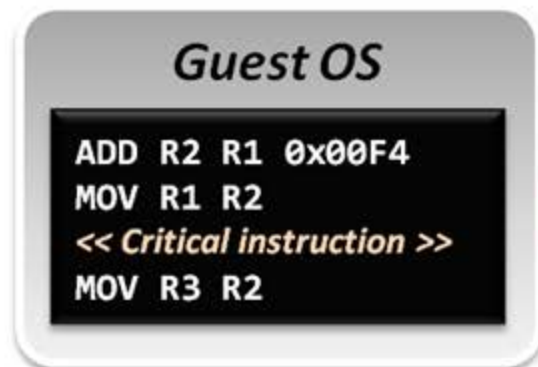
传统OS



VMM和Guest OS

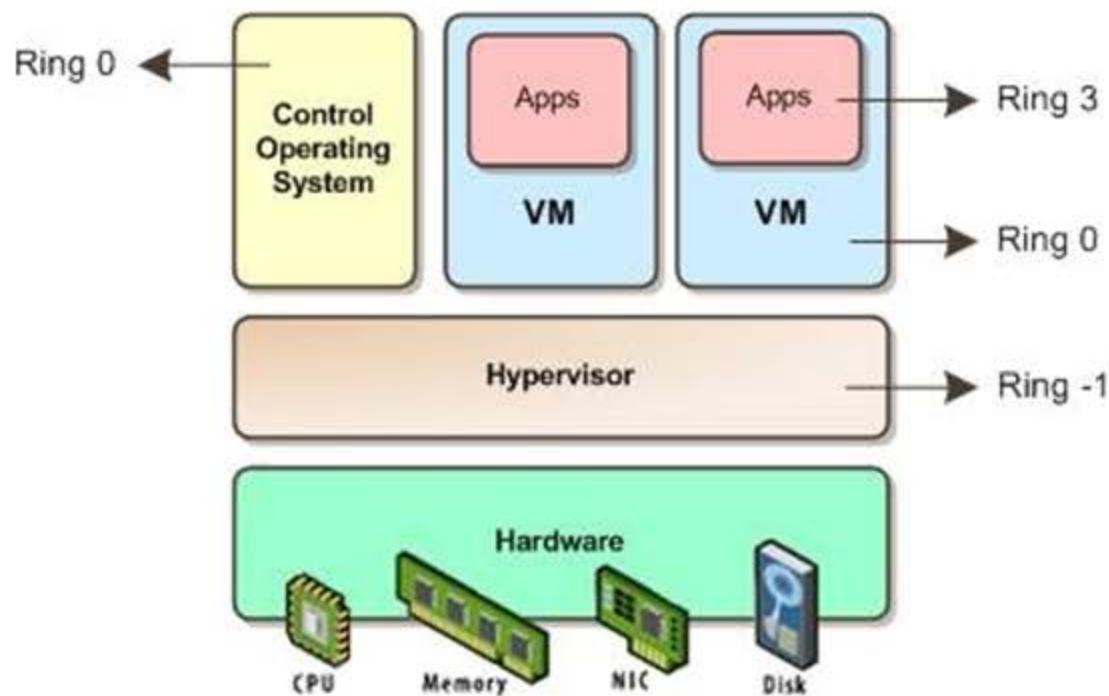
半虚拟化 Para virtualization

- ▶ 也称为：超虚拟化、操作系统辅助虚拟化
- ▶ Hypervisor运行 Ring 0
- ▶ Guest OS不能直接运行在Ring 0，需要对Kernel进行修改，将运行在Ring 0上的，指令转为调用Hypervisor
- ▶ Guest OS上的APP运行在Ring 3
- ▶ 示例：Xen

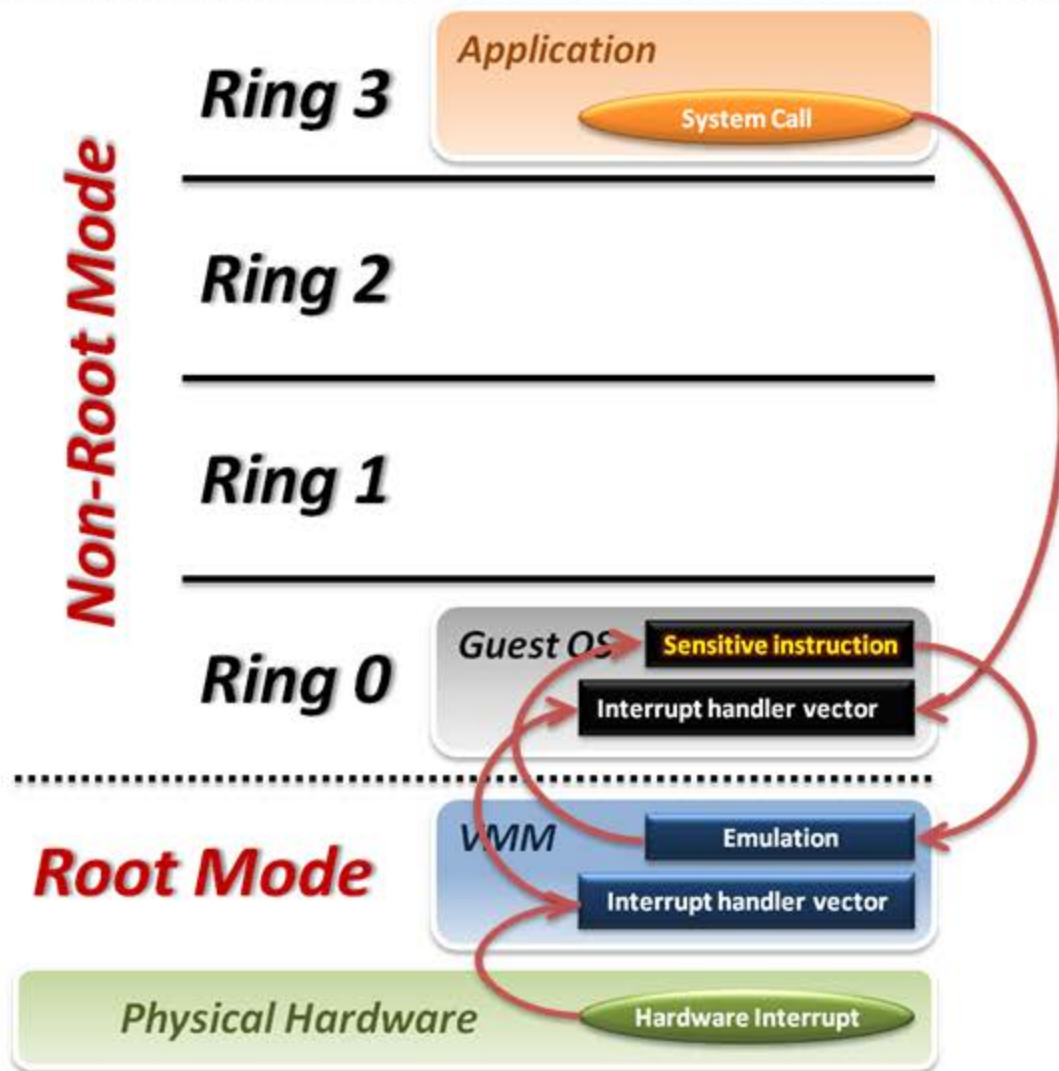


硬件辅助的全虚拟化

- ▶ Full Virtualization with Hardware Assist
- ▶ Intel VT和AMD-V创建一个新的Ring -1单独给Hypervisor使用
- ▶ Guest OS可以直接使用Ring 0而无需修改
- ▶ 示例：
 - ▶ VMware ESXi
 - ▶ Microsoft Hyper-V
 - ▶ Xen3.0
 - ▶ KVM

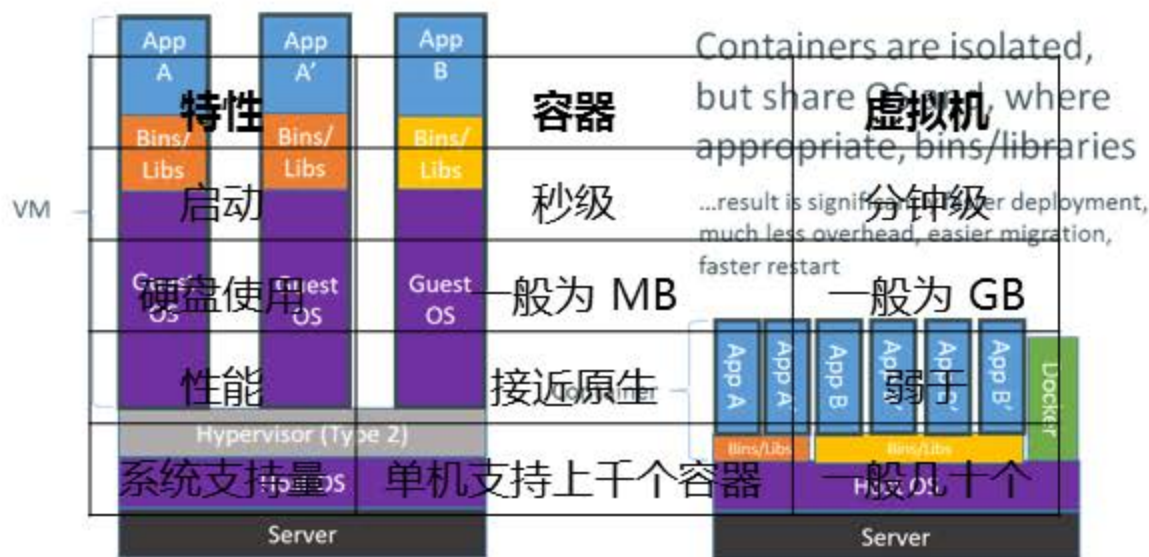


Intel VT-x



LXC和Docker

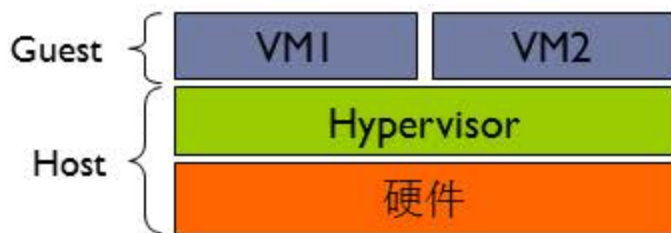
- ▶ 一种轻量级/操作系统虚拟化方式，由Linux内核支持
- ▶ 起源：chroot系统调用，对当前程序及其子进程改变根目录
- ▶ 优势：
 - ▶ 更快速的交付和部署
 - ▶ 更高效的虚拟化
 - ▶ 更轻松的迁移和扩展
 - ▶ 更简单的管理



Hypervisor的分类

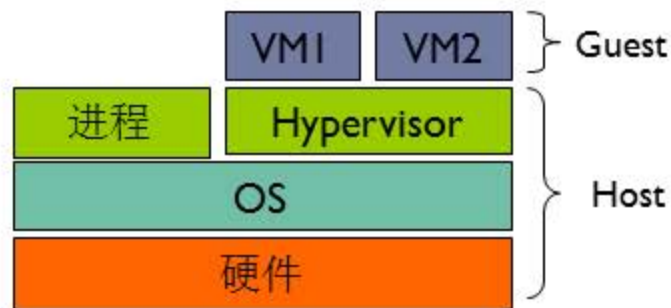
- 1974年，Gerald J. Popek 和Robert P. Goldberg的文章“Formal Requirements for Virtualizable Third Generation Architectures”将Hypervisor分为两类:

类型1：裸金属型
Type 1 (bare-metal)



VMware ESXi, Microsoft Hyper-V, Xen

类型2：宿主型
Type 2 (hosted)



VMware Workstation, Microsoft Virtual PC,
Oracle/Sun VirtualBox, QEMU, KVM

小结：虚拟化技术分类

▶ 虚拟化实现技术：

- ▶ 纯软件仿真: QEMU、Bochs、PearPC
- ▶ 虚拟化翻译
 - ▶ 无硬件辅助的全虚拟化
 - ▶ 半虚拟化
 - ▶ 硬件辅助的全虚拟化
- ▶ 容器技术
 - ▶ LXC和Docker

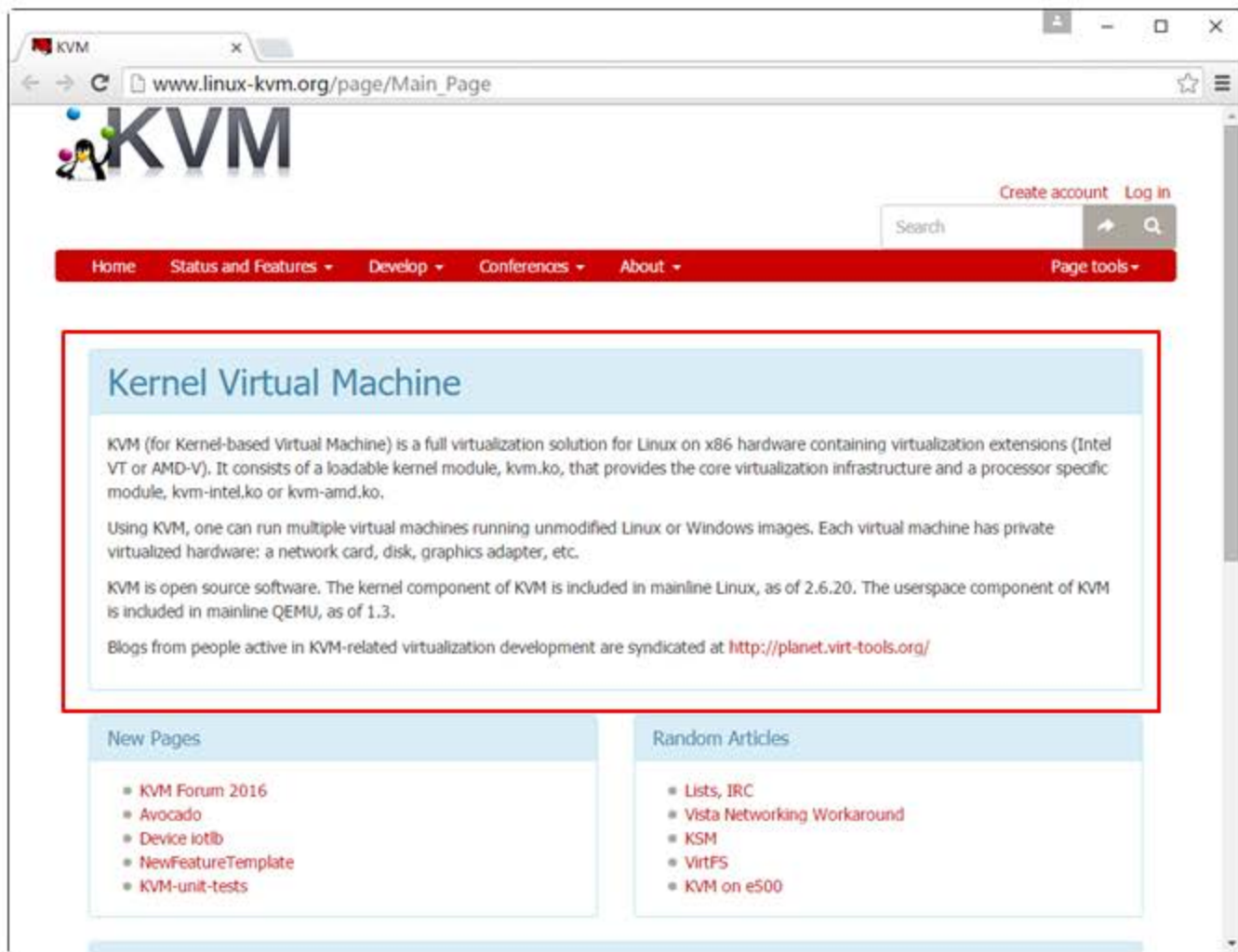
▶ Hypervisor的分类:

- ▶ 类型1：裸金属型
- ▶ 类型2：宿主型

◆ KVM概述

- ▶ KVM历史
- ▶ KVM体系结构
- ▶ QEMU是什么？
- ▶ libvirt是什么？
- ▶ KVM集中管理与控制
- ▶ 扩展知识
 - ▶ Hypervisor /VMM 的选择
 - ▶ 国产虚拟化解决方案概述

KVM是什么？



KVM是什么？

- ▶ <http://www.linux-kvm.org>
- ▶ KVM (for Kernel-based Virtual Machine) is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (Intel VT or AMD-V). It consists of a loadable kernel module, `kvm.ko`, that provides the core virtualization infrastructure and a processor specific module, `kvm-intel.ko` or `kvm-amd.ko`.
- ▶ Using KVM, one can run multiple virtual machines running unmodified Linux or Windows images. Each virtual machine has private virtualized hardware: a network card, disk, graphics adapter, etc.
- ▶ KVM is open source software. The kernel component of KVM is included in mainline Linux, as of 2.6.20. The userspace component of KVM is included in mainline QEMU, as of 1.3.

KVM的历史

- ▶ 以色列的创业公司Qumranet创建了KVM
 - ▶ 早期的主要开发者Avi Kivity
 - ▶ 2006年10月，在完成基本功能、动态迁移、主要的性能优化后，正式宣布了KVM的诞生
 - ▶ 2007年2月发布的内核2.6.20中，开始正式包括了KVM
- ▶ 2008年9月，Redhat 1.7亿美收购
 - ▶ RHEL 5.4，在集成Xen的基础上，又将KVM添加进来
 - ▶ 2011年11月，RHEL6使用KVM彻底替换了Xen
- ▶ QEMU
 - ▶ QEMU是一个通用的开源的硬件模拟器，可以模拟多种硬件
 - ▶ QEMU-KVM从分支到与主干合并，QEMU成立KVM在用户空间的管理工具



KVM体系结构

▶ KVM

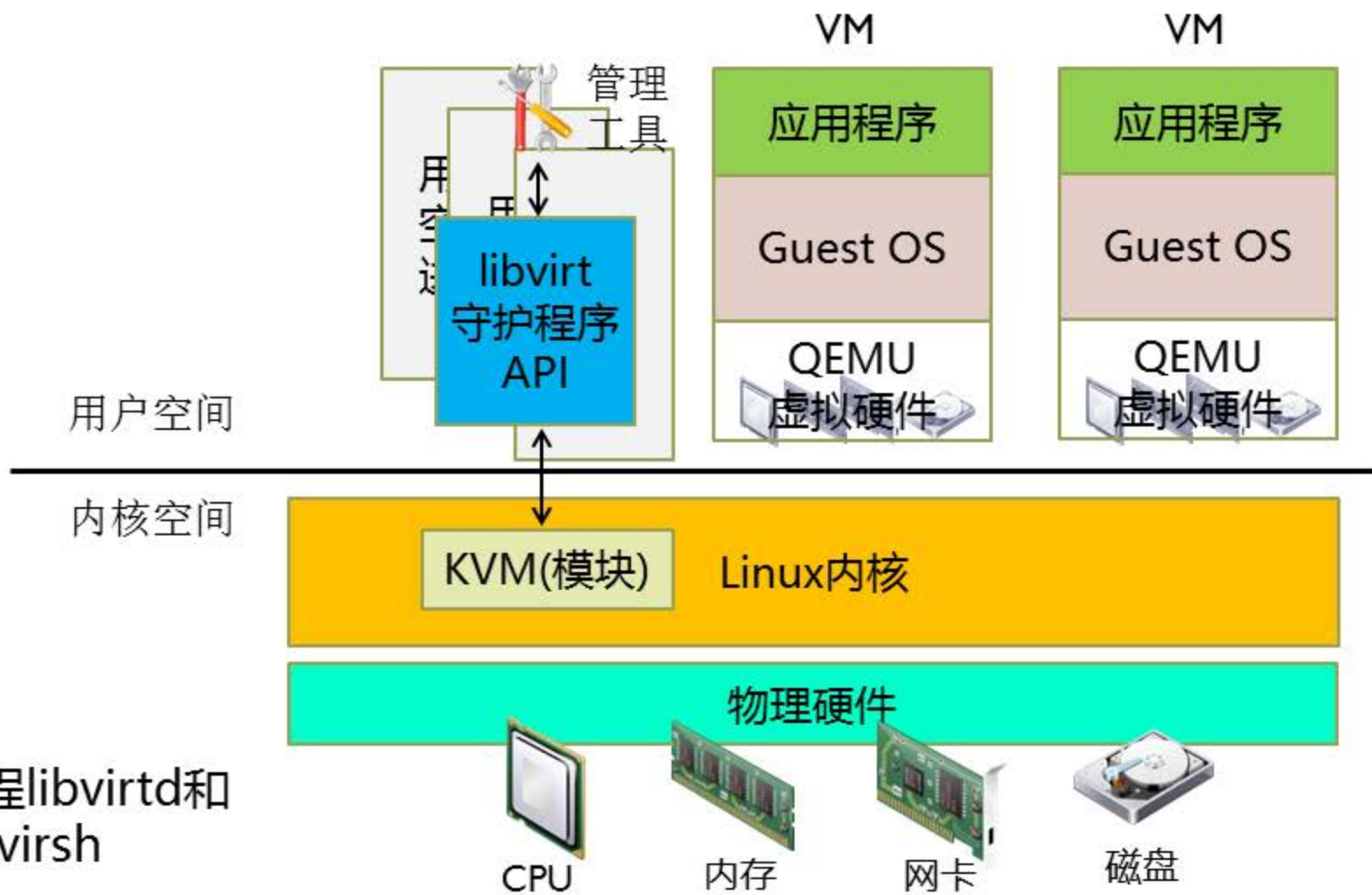
- ▶ 初始化CPU硬件，打开虚拟化模式，以支持虚拟机的运行。
- ▶ 负责CPU、内存、中断控制器、时钟

▶ QEMU

- ▶ 模拟网卡、显卡、存储控制器和硬盘

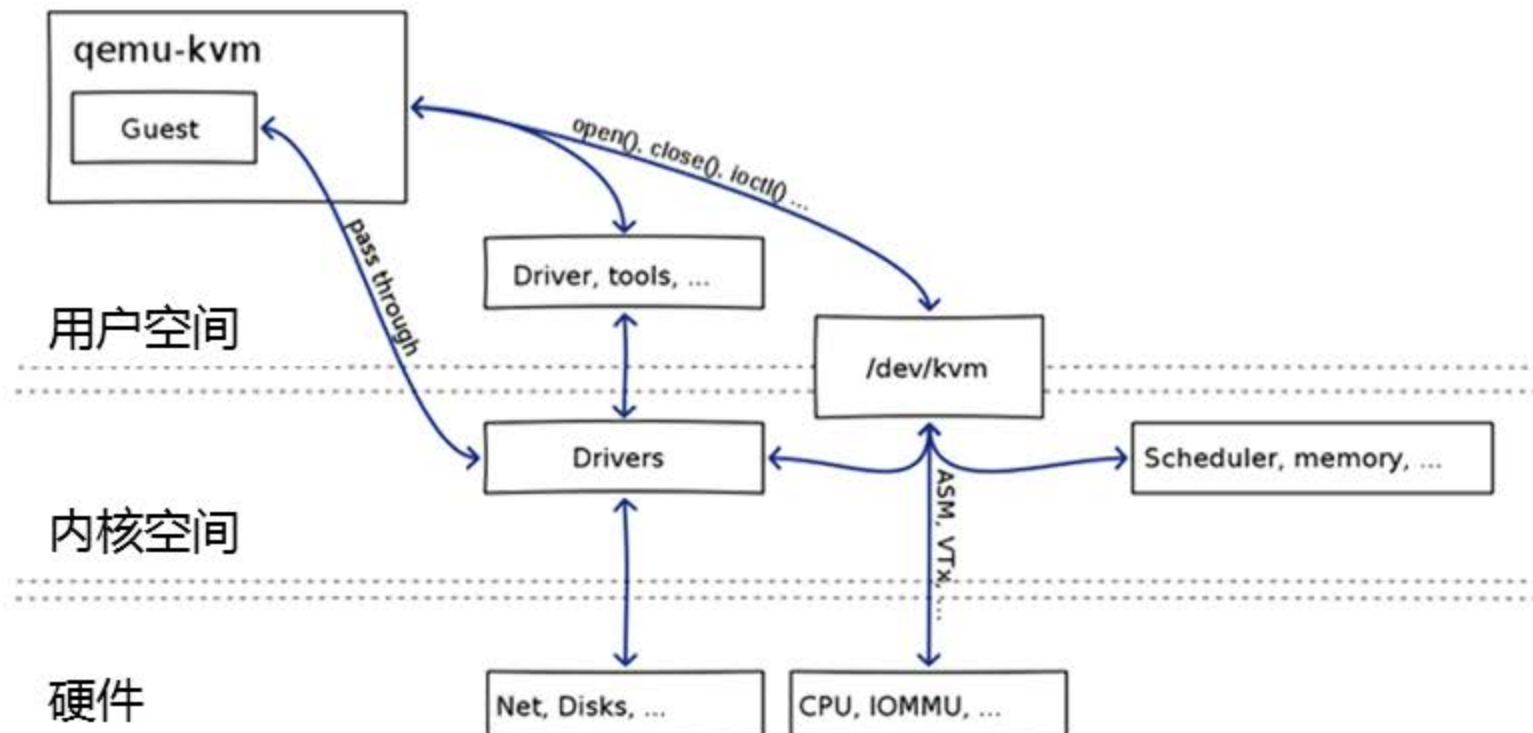
▶ libvirt

- ▶ 它提供统一API、守护进程libvirtd和一个默认命令行管理工具virsh

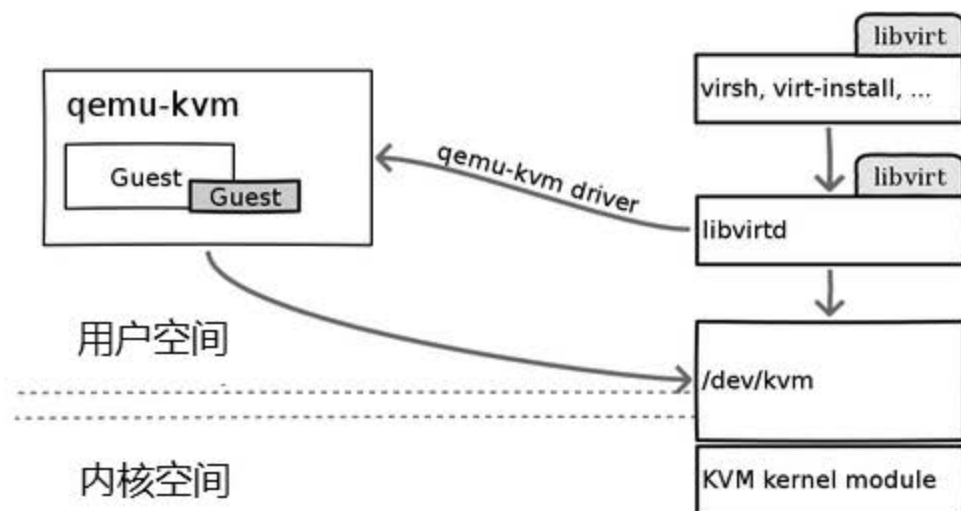
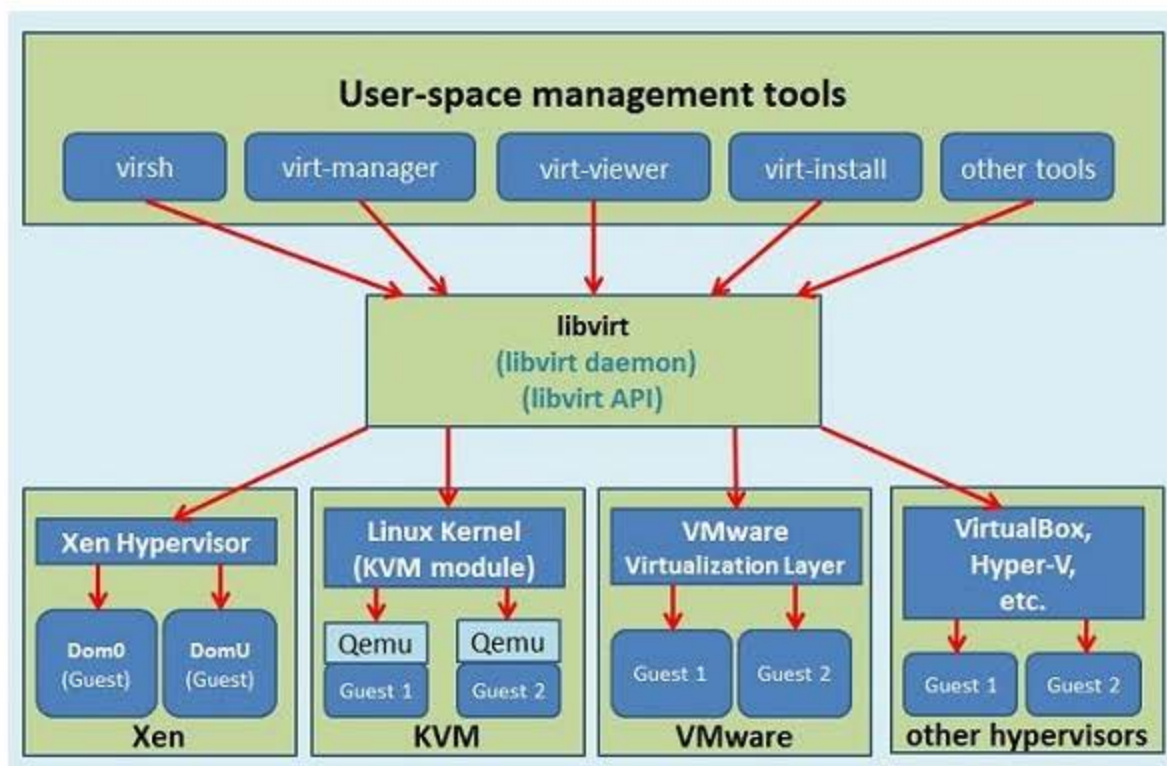


Qemu与KVM

- ▶ Qemu通过ioctl调用/dev/kvm接口，将有关CPU指令的部分交由KVM来做
- ▶ Qemu还是模拟网卡、显卡、存储控制器和硬盘等，会影响这些设备的性能
- ▶ 通过pass through半虚拟化设备virtio_blk, virtio_net，提高设备性能



libvirt与KVM



KVM集中管理与控制

▶ http://www.linux-kvm.org/page/Management_Tools

▶ 示例：

▶ Ovirt

▶ 功能强大，是Redhat 虚拟化管理平台RHEV的开源版本。

▶ <http://www.ovirt.org/>

▶ WebVirtMgr

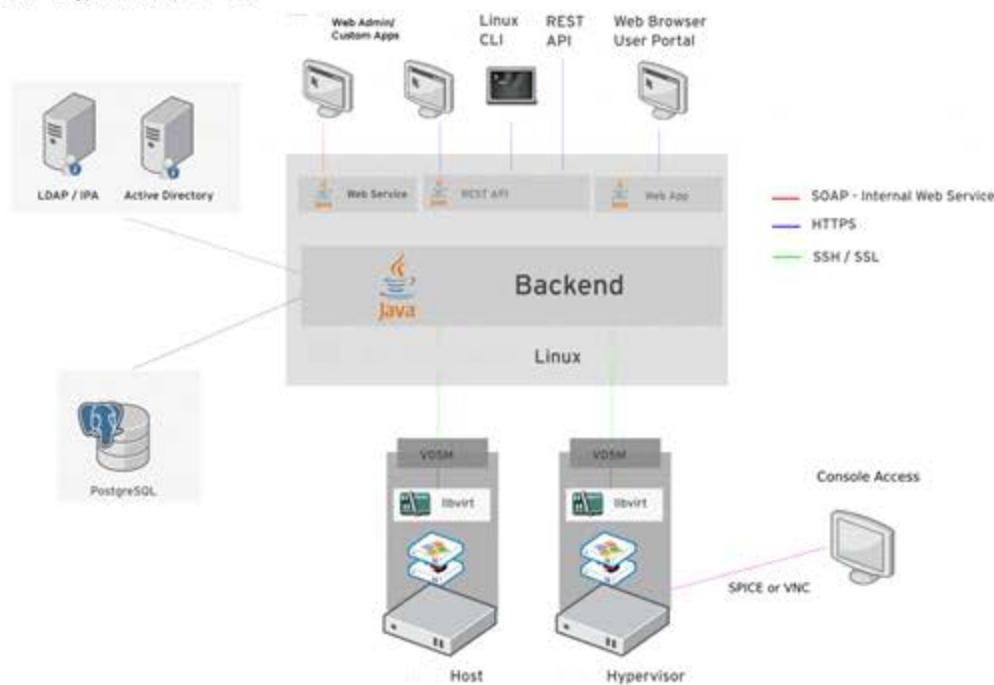
▶ <https://www.webvirtmgr.net>

▶ virt-manager的Web模式的替代品

▶ ConVirt

▶ <http://www.convirture.com/>

▶ 分开源版本和商业版本，



Hypervisor /VMM 的选择

- ▶ 电信/ISP 公共云
 - ▶ 单一hypervisor
 - ▶ KVM
- ▶ 互联网公司
 - ▶ 单一hypervisor
 - ▶ KVM
- ▶ Web托管和SAS细分市场
 - ▶ 单一或多个hypervisor
 - ▶ 容器 (LXC, Parallels, Docker)
 - ▶ KVM
- ▶ 企业
 - ▶ 多种Hypervisor
 - ▶ KVM 和 vCenter/ESXi

国产虚拟化解决方案概述

- ▶ Stop Trying to Reinvent the Wheel , 不要重复造轮子
- ▶ 国产虚拟化解决方案重点在虚拟化的管理与桌面协议
- ▶ Hypervisor/VMM选择 (按汉语拼音排序☺) :
 - ▶ H3 : KVM
 - ▶ 红山 : Xen
 - ▶ 华为 : Xen → KVM → Xen
 - ▶ 浪潮 : Xen
 - ▶ 普华 : KVM
 - ▶ 锐捷 : KVM
 - ▶ 深信服 : KVM
 - ▶ ... 其他KVM居多

◆ 资源与参考资料

▶ Redhat官方文档

- ▶ <https://access.redhat.com/documentation>
- ▶ Virtualization Getting Started Guide
- ▶ Virtualization Deployment and Administration Guide
- ▶ Virtualization Tuning and Optimization Guide
- ▶ Virtualization Security Guide

▶ KVM 主页

- ▶ <http://www.linux-kvm.org/>
- ▶ 提供了 KVM 的详细信息，以及大量信息来源和相关站点的链接

▶ libvirt Virtualization API 网站

- ▶ <http://www.libvirt.org/>
- ▶ 提供了有关该 API、所支持的虚拟化抽象及其使用的 XML 格式的详细信息。

总结

- ▶ 虚拟化概述
- ▶ KVM概述
- ▶ 资源与参考资料