

- 云计算
 - 虚拟化模型
 - KVM架构
 - KVM模块
 - QEMU设备模型
 - Intel虚拟化技术

云计算

虚拟化模型



图 2-1 虚拟化模型

KVM架构

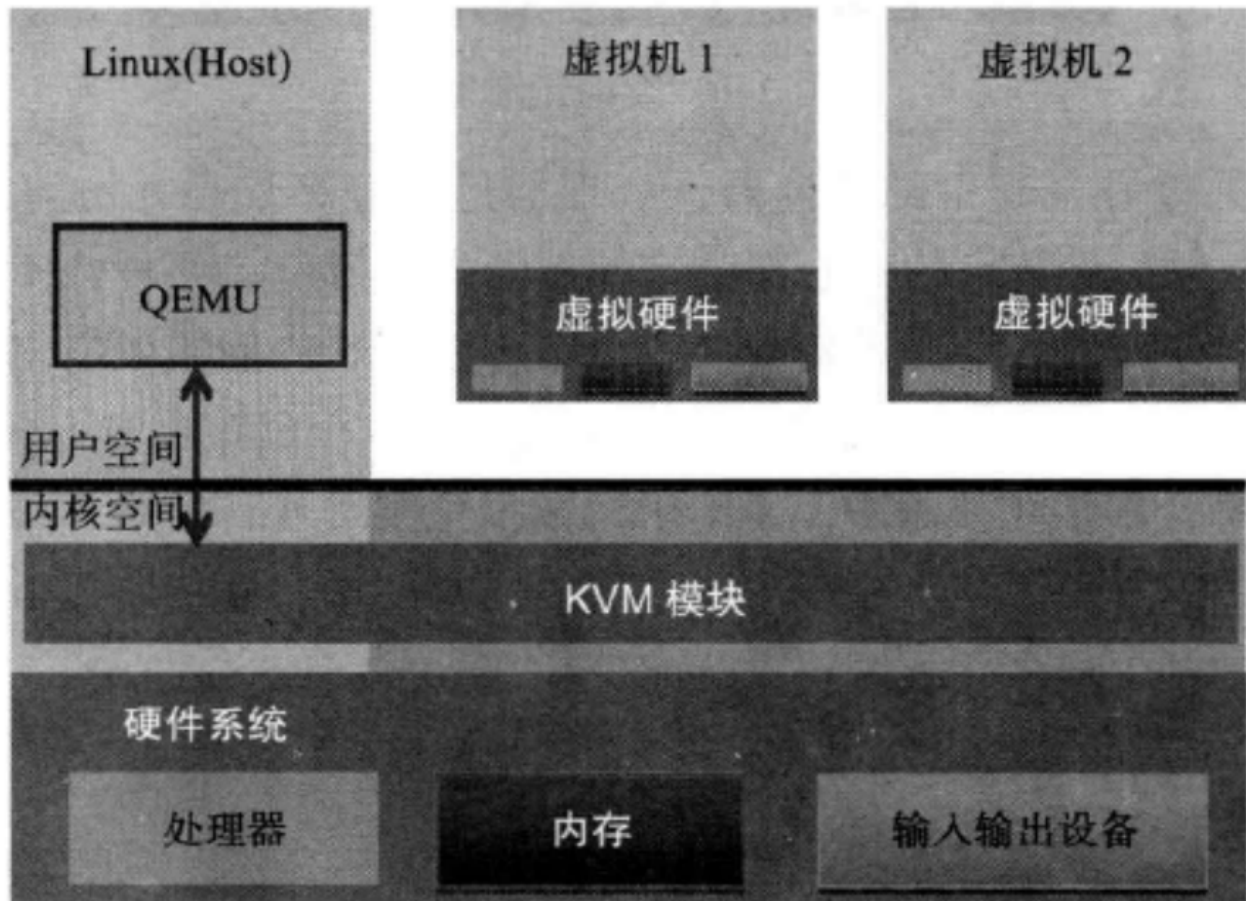


图 2-2 KVM 架构

KVM模块

处理器虚化

内存虚化

QEMU设备模型

其它虚化

Intel虚拟化技术

硬件虚化支持

- VT-x CPU
- VT-d 芯片
- VMDq Intel网卡
- SR-IOV PCI组织定义的单根设备虚拟化协议

Intel® Virtualization Technology Evolution

Vector 3:

IO Device Focus

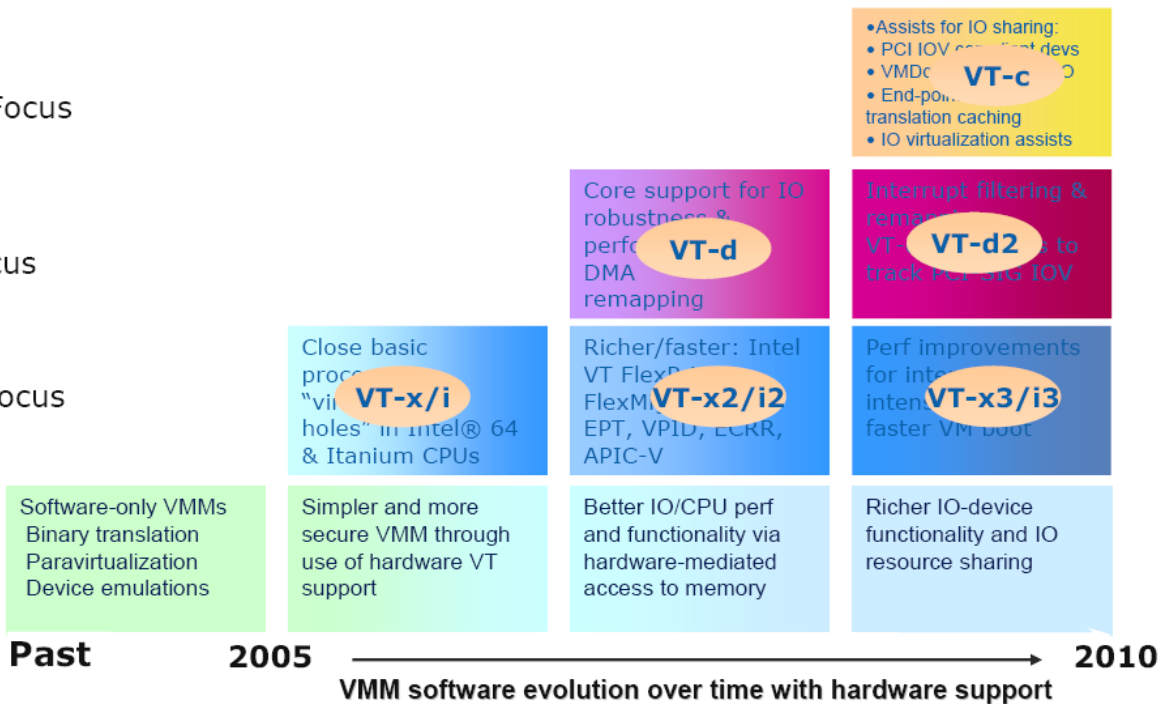
Vector 2:

Chipset Focus

Vector 1:

Processor Focus

VMM
Software
Evolution



• VT-x

[运行ESXI上的64bit Guest OS基本指令]

Intel运用Virtualization虚拟化技术中的一个指令集。VT-x有助于提高基于软件的虚拟化解决方案的灵活性与稳定性。通过按照纯软件虚拟化的要求消除虚拟机监视器(VMM)代表客户操作系统来听取、中断与执行特定指令的需要,不仅能够有效减少VMM干预,还为VMM与客户操作系统之间的传输平台控制提供了有力的硬件支持,这样在需要VMM干预时,将实现更加快速、可靠和安全的切换。

• VT-d

[虚拟机可以直接针对周边硬体做存取,由北桥晶片来支援及BIOS来开启]

简单来说,VT-d技术可以使其中一个Guest OS可以直接存取硬体设备,例如Raid Card,Lan Card,Display Card。假设Display Card pass through 到Guest OS后,只有这个Guest OS可以控制及使用。事实上在虚拟环境中,还是有许多直接硬体存取的机会,如备份伺服器,常常需要直接存取HBA Card才能加快速度,此时VT-d就大派用场。

• VT-c

[架构都是以Network为主,要由I/O装置来支援]

◦ Virtual Machine Direct Connect

虚拟机上的虚拟网络卡传送主要透过VMM(或是hypervisor)来进行传输,而VMDc允许VM可以直接针对实体网路I/O进行存取。虽然在Intel VT-d技术中已经允许虚拟机可以直接和实体I/O连结,不过VMDc使用了PCI-SIG的Single Root I/O Virtualization的技术。透过这项技术更进一步提升VT-d的功能。它可以同时让多个虚拟机与实体I/O装置同时建立通道!

◦ Virtual Machine Device Queues

目前我们使用的虚拟化不管是RHEL的KVM或是Xen还是VMware/Citrix,在处理虚拟机的网络卡时都会透过一个由VMM(或是叫Hypervisor)所管理的虚拟化Switch,而这Switch主要的功能就是转送封包给正确的虚拟机,但哪一些封包要流向哪一个虚拟机都需要额外的CPU资源才能完成这些工作。而Intel所使用的VMDq就是为了减少这些额外CPU的处理,透过网卡晶片内建的Layer2 classifier / sorter以加速网路资料的传送,它可以先行将不同的虚拟机所需

的封包,直接在晶片里面安排好再透过**receive queue**,直接给虚拟机.这样就不需再透过**Virtual switch**转送封包.!!大大减少网路的负载与**CPU**的使用率!
