

KVM基础功能

支持

- 硬件支持 VT-x VT-d
- 系统支持 kernel > 3.5

计算机系统的子系统

- CPU 处理器
- Memory 内存
- Storage 存储
- Network 网络
- Display 显示

CPU

vCPU (virtual CPU)

进程模式

- 用户模式 (User Mode) 主要处理I/O的模拟和管理，由QEMU实现。
- 内核模式 (Kernel Mode) 主要处理高性能和高安全的指令。
- 客户模式 (Guest Mode) 主要执行Guest的大部分命令。

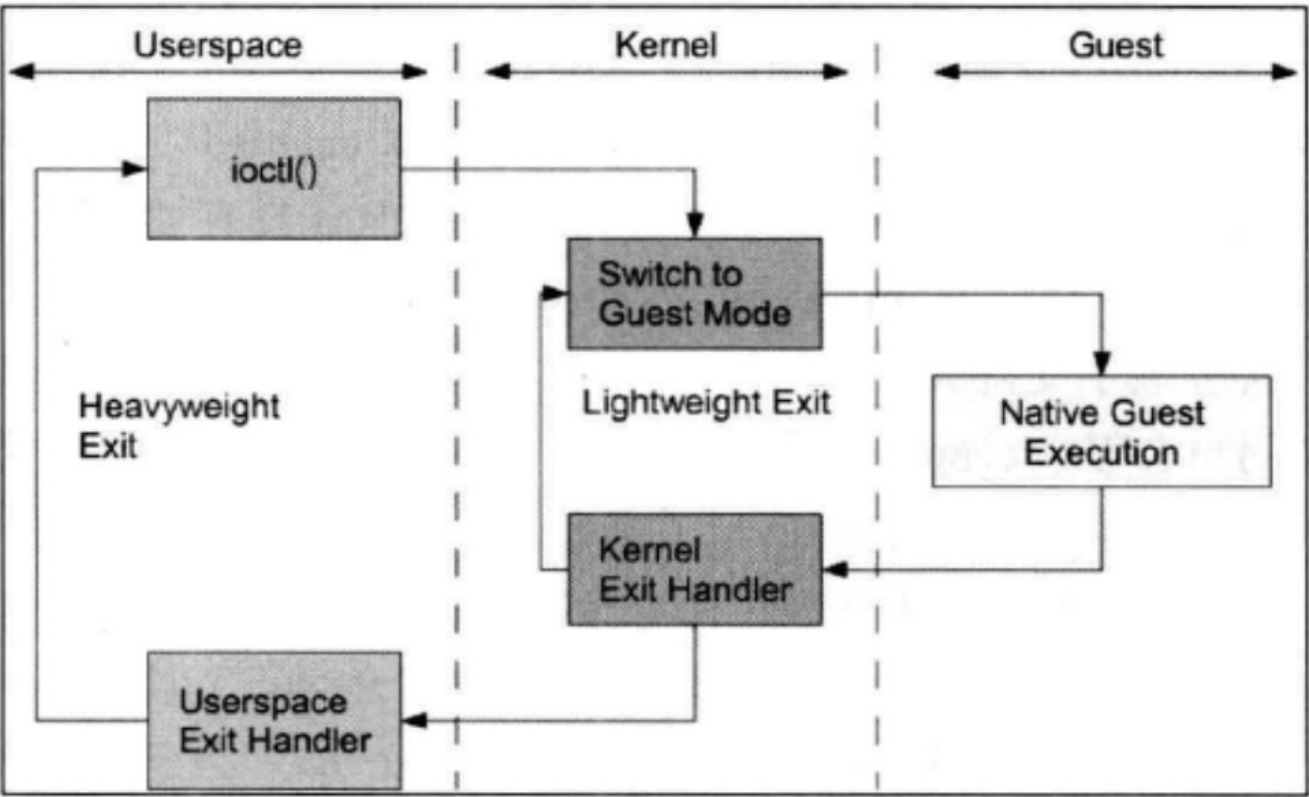


图 4-1 vCPU 在 KVM 中的三种执行模式

- kvm

- kvm_intel

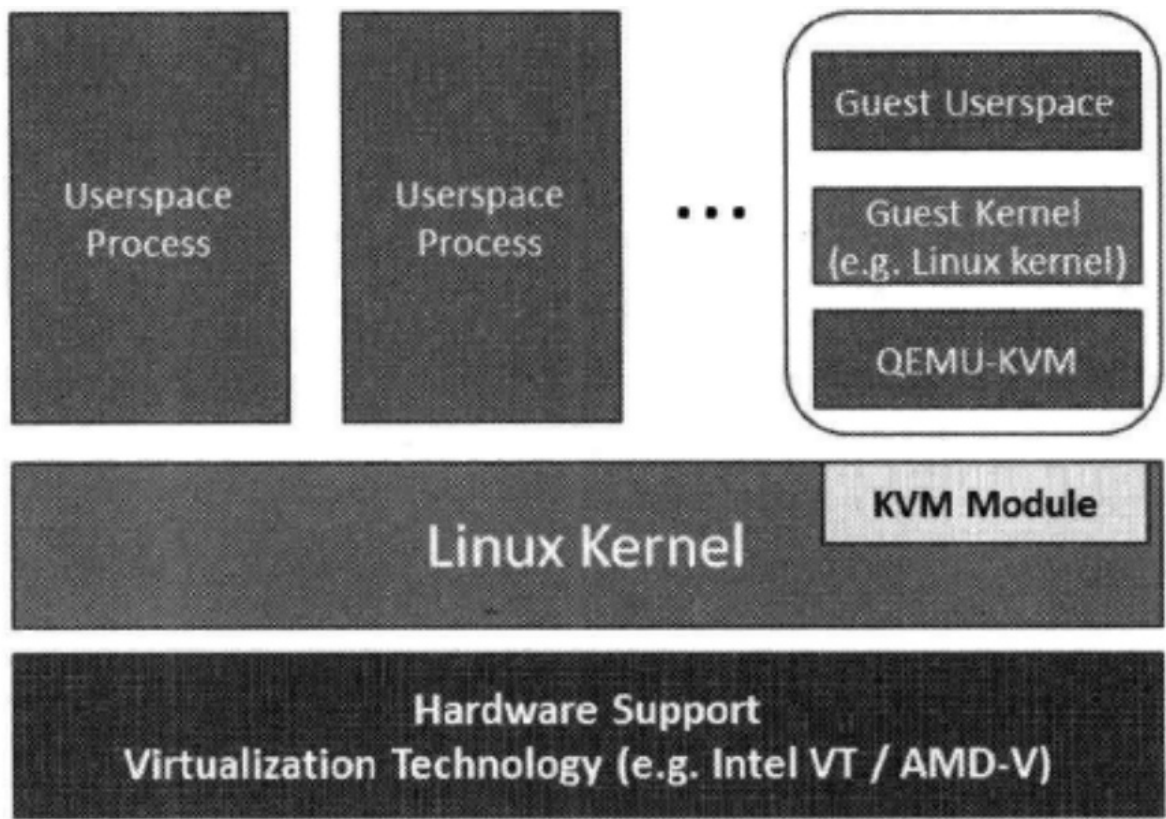


图 4-2 KVM 系统的分层架构

SMP

SMP (Symmetric Multi-Processor) 对称多处理器

```
@import "Linux_CPU-Topology.md";
```

CPU 超配

over-commit 过载使用

CPU 模型

- qemu64
- qemu32
- 直接传递
- 其它POWER

VMM自定义策略的CPU类型，可以提供一些高级的过滤功能，同一组硬件平台上的动态迁移更加平滑和安全

```
more /usr/share/libvirt/cpu_map.xml | grep "model name"
```

```
<domain type='kvm'>
  <cpu mode='custom' match='exact'>
    <model fallback='allow'>Westmere</model>
  </cpu>
</domain>
```

```
<domain type='kvm'>
```

```
<cpu mode='host-passthrough' />
<domain type='kvm'>
```

- 物理CPU特性供虚拟机使用,嵌套
- 公有云,客户看到物理CPU,体验会比较好
- 虚拟机的迁移会有问题

进程的处理亲和性和vCPU绑定

进程的处理亲和性（*Processor Affinity*），即CPU的绑定设置，是指将进程绑定到特定的一个或多个CPU上去执行，而不允许将进程调度到其他的CPU上。

```
grub add
isolcpus=1,2
```

```
ps -elo psr | grep 0 | wc -l
ps -elo psr | grep 1 | wc -l
ps -elo psr | grep 2 | wc -l
ps -elo psr | grep 3 | wc -l
```

```
ps -elo ruser,pid,ppid,lwp,psr,args | awk '{if($5==1) print $0}'
ps -elo ruser,pid,ppid,lwp,psr,args | awk '{if($5==2) print $0}'
```

migration 用于进程在没CPU间迁移 kworker 用于处理workqueues ksoftirqd 用于调度CPU软中断的进程

-e 显示所有进程 -L 显示LWP light weight process -o 自定义输出

psr 处理器编号 lwp 线程ID ruser 运行进程的用户 pid 进程ID ppid 父进程ID args 运行的命令和参数

```
virt-install --name=c1 --disk path=/kvm/c1.img --memory=1024 --vcpus=2 --graphics spice --location=/kvm/CentOS.iso
ps -elo ruser,pid,ppid,lwp,psr,args | grep qemu | grep -v grep
```

```
taskset --help
taskset -pc 01 5339
```

```
virsh vcpuinfo centos7.0
virsh vcpupin centos7.0 0 20
virsh vcpupin centos7.0 1 21
virsh vcpupin centos7.0 2 22
virsh vcpupin centos7.0 3 23
```

```
Ctrl + Alt + 2
info cpus
```