

分享地理价值 设计共同未来



虚拟化和KVM技术探讨

左伟
2011-5-6



主要内容

一、虚拟化简介

二、KVM技术介绍

三、在RHEL5.6上的实际部署安装说明



一、虚拟化简介

1.虚拟化概述

2.虚拟化的分类

3.服务器虚拟化

1.虚拟化概述

虚拟化是一个广义的术语，在计算机方面通常是指计算元件在虚拟的基础上而不是真实的基础上运行。虚拟化技术可以扩大硬件的容量，简化软件的重新配置过程。CPU的虚拟化技术可以单CPU模拟多CPU并行，允许一个平台同时运行多个操作系统，并且应用程序都可以在相互独立的空间内运行而互不影响，从而实现充分利用计算机资源并显著提高计算机的工作效率。

1.虚拟化概述

- “虚拟化是表示计算机资源的抽象方法，通过虚拟化可以用与访问抽象前资源一致的方法访问抽象后的资源。这种资源的抽象方法并不受实现、地理位置或底层资源的物理配置的限制。”

——Wikipedia, 维基百科

- “虚拟化是为某些事物创造的虚拟（相对于真实）版本，比如操作系统、计算机系统、存储设备和网络资源等。”

——WhatIs.com, 信息技术术语库

- “虚拟化是为一组类似资源提供一个通用的抽象接口集，从而隐藏属性和操作之间的差异，并允许通过一种通用的方式来查看并维护资源。”

——Open Grid Services Architecture

1.虚拟化概述

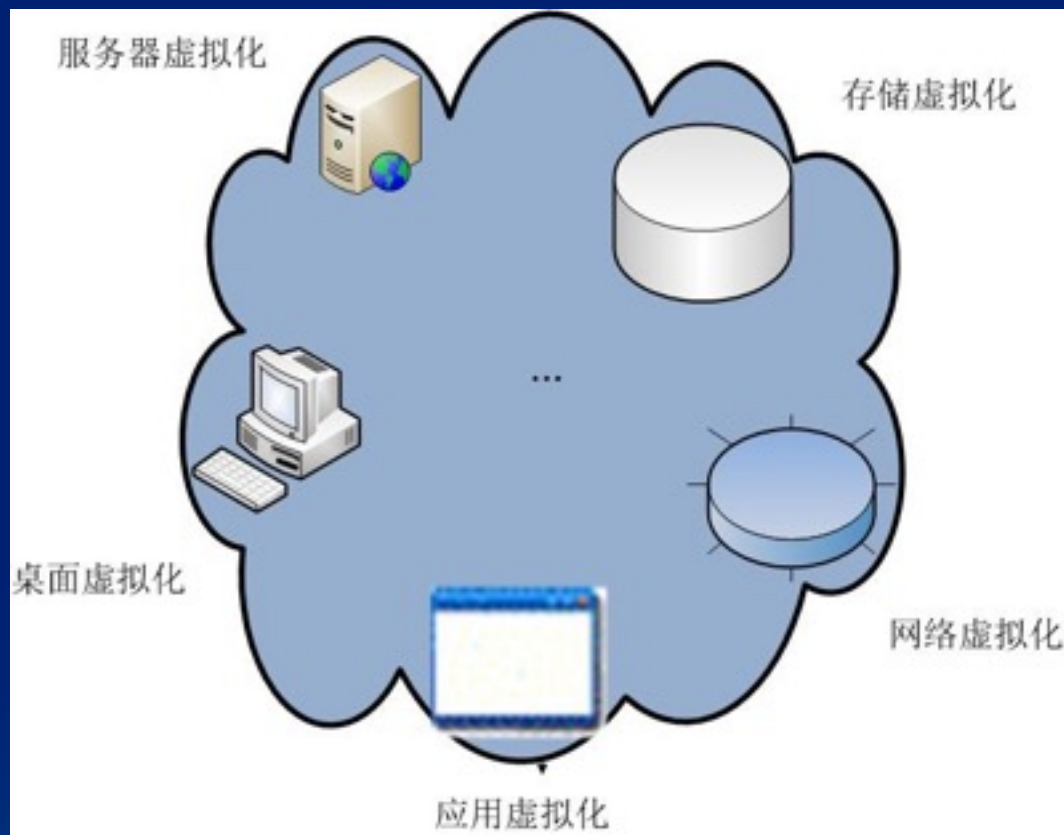
尽管以上几种定义表述方式不尽相同，但仔细分析一下，不难发现它们都阐述了三层含义：

- 虚拟化的对象是各种各样的资源；
- 经过虚拟化后的逻辑资源对用户隐藏了不必要的细节；
- 用户可以在虚拟环境中实现其在真实环境中的部分或者全部功能。

1.虚拟化概述

虚拟化是资源的逻辑表示，它不受物理限制的约束。

--IBM公司



1.虚拟化概述

起步阶段

沉淀阶段

爆发阶段

1959: 克里斯托弗
(Christopher Strachey)
"大型高速计算机
中的时间共享"

1965: IBM
"M44/44X"
虚拟内存
管理机制

Iptunnel
VPN

1995: JAVA
第一个
开发工具
JDK1.0alpha

2006:
Intel VT-d
AMD-V™

2011: IBM
Power7

1950

1960

1970

1980

1990

2000

2010

.....

1965:
IBM
IBM7044
第一台
商用机
大型机

1978: IBM
RAID
(冗余独立
磁盘阵列)

1999:
Vmware
X86平台
虚拟化软件

2008: MS
Hyper-V
Virtual Server

2.虚拟化分类

从实现层次进行划分：

- 资源虚拟化Resource Virtualization
- 平台虚拟化Platform Virtualization
- 应用程序虚拟化Application Virtualization

2.虚拟化分类

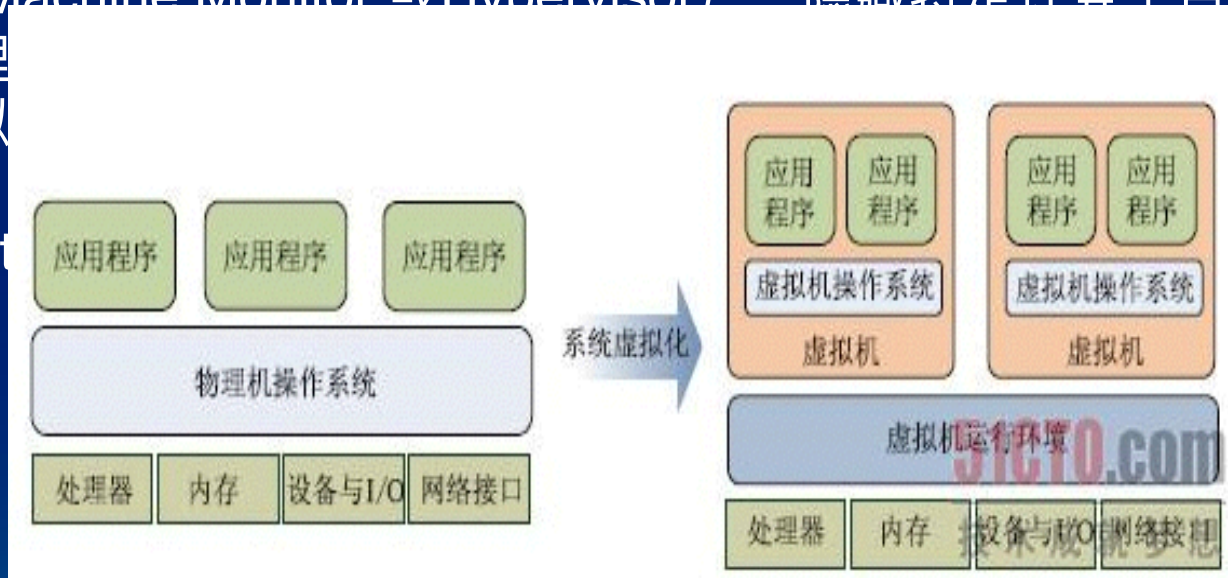
资源虚拟化：也可以称之为基础设施虚拟化，指网络、存储和文件系统等基础设施的虚拟化。

- 网络虚拟化是指将网络的硬件和软件资源整合，向用户提供虚拟网络连接的虚拟化技术。比如虚拟局域网（Virtual LAN, VLAN），虚拟专用网（Virtual Private Network, VPN）等。
- 存储虚拟化是指为物理的存储设备提供一个抽象的逻辑视图，用户可以通过这个视图中的统一逻辑接口来访问被整合的存储资源。比如磁盘阵列技术（Redundant Array of Inexpensive Disks, RAID），网络附加存储（Network Attached Storage, NAS）和存储区域网（StorageArea Network, SAN）等。

2.虚拟化的分类

平台虚拟化，就是针对计算机和操作系统的虚拟化。比如使用虚拟化软件在一台物理机上虚拟出一台或多台虚拟机。这也是我们通常意义上的虚拟化。

平台虚拟化通过使用控制程序（Control Program，也被称为 Virtual Machine Monitor 或 Hypervisor）隐藏特定计算平台的实际物理资源，为上层应用提供一个或多个虚拟化的计算平台（称为虚拟操作系统（Guest OS）或虚拟机（VM））。



2.虚拟化分类

平台虚拟化又可以细分如下几种子类：

- a. 全虚拟化， Full Virtualization： 比较著名的全虚拟化 VMM 有 Microsoft Virtual PC、VMware Workstation、Sun Virtual Box、Parallels Desktop for Mac 、 QEMU和KVM。
- b. 超虚拟化， Paravirtualization： Xen
- c. 硬件辅助虚拟化， Hardware-Assisted Virtualization： Intel-VT 和 AMD-V
- d. 部分虚拟化， Partial Virtualization： 过渡
- e. 操作系统级虚拟化， Operating System Level Virtualization： Solaris Container, FreeBSD Jail 和 OpenVZ等

2.虚拟化分类

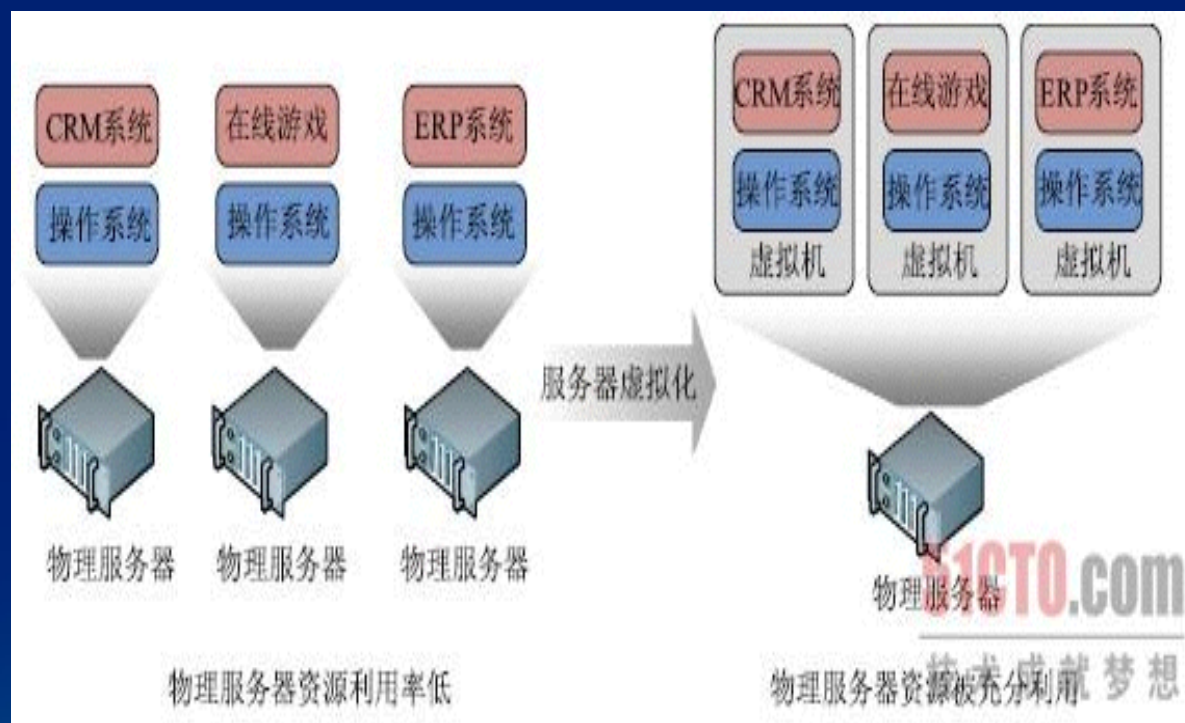
软件虚拟化：主要包括应用虚拟化和高级语言虚拟化。

应用虚拟化将应用程序与操作系统解耦合，为应用程序提供了一个虚拟的运行环境。在这个环境中，不仅包括应用程序的可执行文件，还包括它所需要的运行时环境。当用户需要使用某款软件时，应用虚拟化服务器可以实时地将用户所需的程序组件推送到客户端的应用虚拟化运行环境。当用户完成操作关闭应用程序后，他所做的更改和数据将被上传到服务器集中管理。这样，用户将不再局限于单一的客户端，可以在不同的终端上使用自己的应用，比如云端。

高级语言虚拟化解决的是可执行程序在不同体系结构计算机间迁移的问题。在高级语言虚拟化中，由高级语言编写的程序被编译为标准的中间指令。这些中间指令在解释执行或动态翻译环境中被执行，因而可以运行在不同的体系结构之上，比如Java虚拟机。

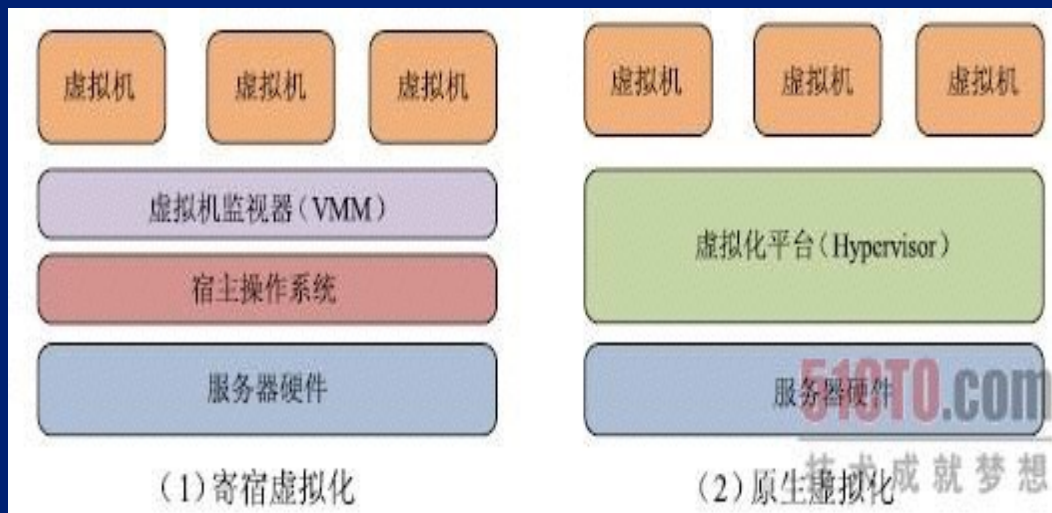
3.服务器虚拟化

服务器虚拟化：将平台虚拟化技术应用于服务器上，将一个服务器虚拟成若干个服务器使用。



3.服务器虚拟化

典型实现



- 虚拟机监视器提供

硬件资源抽象，

	寄宿虚拟化	原生虚拟化
是否依赖于宿主操作系统	完全	不
性能	低	高
实现的难易程度	易	难

为各操作系统提供运行环境。

- 虚拟化平台（**Hypervisor**）：负责虚拟机的托管和管理。它直接运行在硬件之上，因此其实现直接受底层体系结构的约束。

3.服务器虚拟化

下面是几种使用最广泛的服务器虚拟化产品。

- Citrix公司的Xen。
- IBM公司的PowerVM、zVM。
- Microsoft公司的Virtual PC、Virtual Server和Hyper-V。
- VMware公司的VMware Server、VMware Workstation、VMwarePlayer和VMware ESX Server。

3.服务器虚拟化

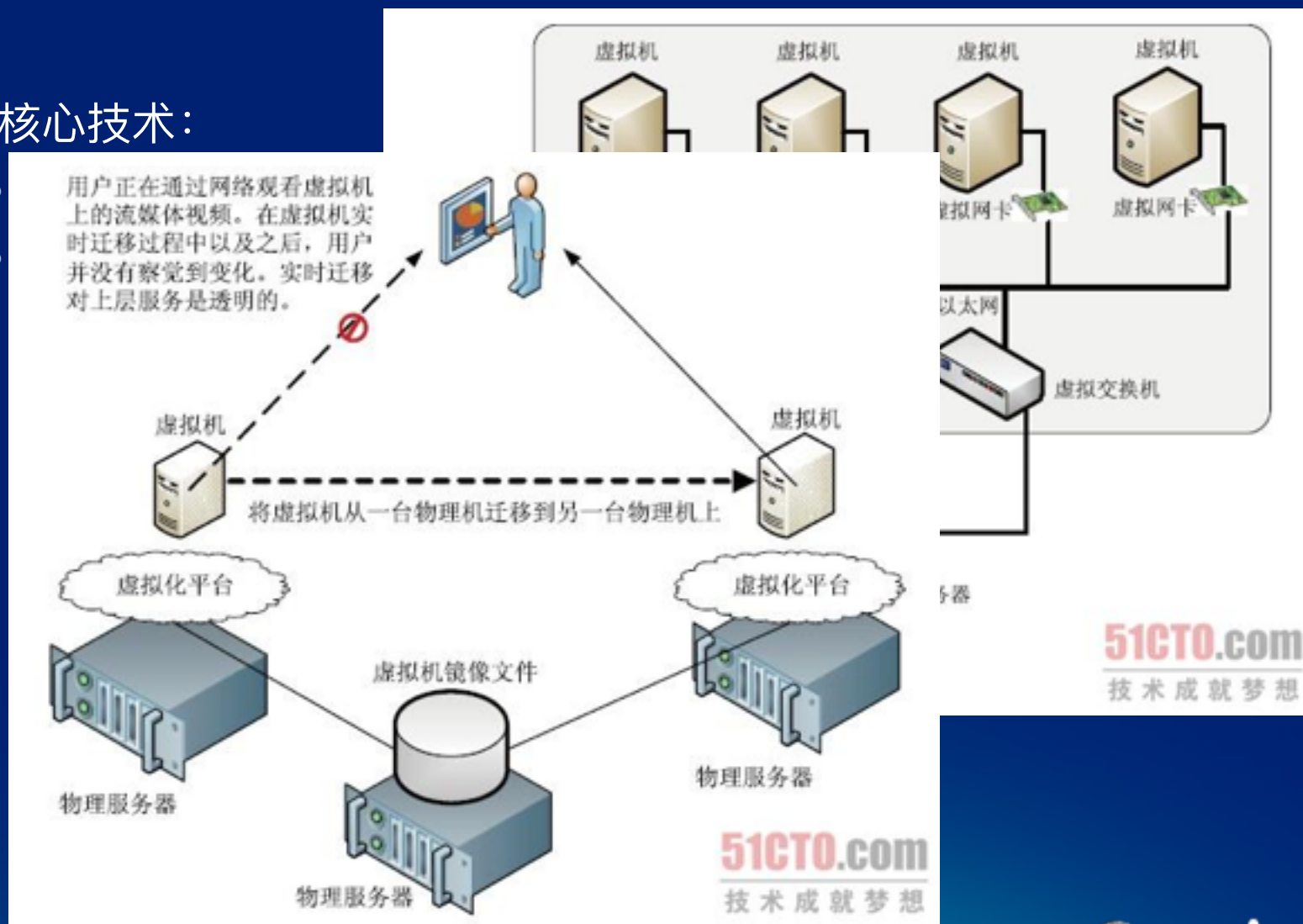
关键特性：

- 多实例--通过服务器虚拟化，在一个物理服务器上可以运行多个虚拟服务器，即可以支持多个客户操作系统。
- 隔离性--在多实例的服务器虚拟化中，一个虚拟机与其他虚拟机完全隔离。
- 封装性--也即硬件无关性，便于在不同的硬件间备份、移动和复制等，也保证了虚拟机的兼容性。
- 高性能--服务器虚拟化的高性能是指虚拟机监视器的开销要被控制在可承受的范围之内。

3.服务器虚拟化

核心技术:

- 用户正在通过网络观看虚拟机上的流媒体视频。在虚拟机实时迁移过程中以及之后，用户并没有察觉到变化。实时迁移对上层服务是透明的。
-



3.服务器虚拟化

技术优势：

- 降低运营成本
- 提高应用兼容性
- 加速应用部署
- 提高服务可用性
- 提升资源利用率
- 动态调度资源—实时迁移
- 降低能源消耗



二、全虚拟化解决方案--KVM

1. KVM介绍

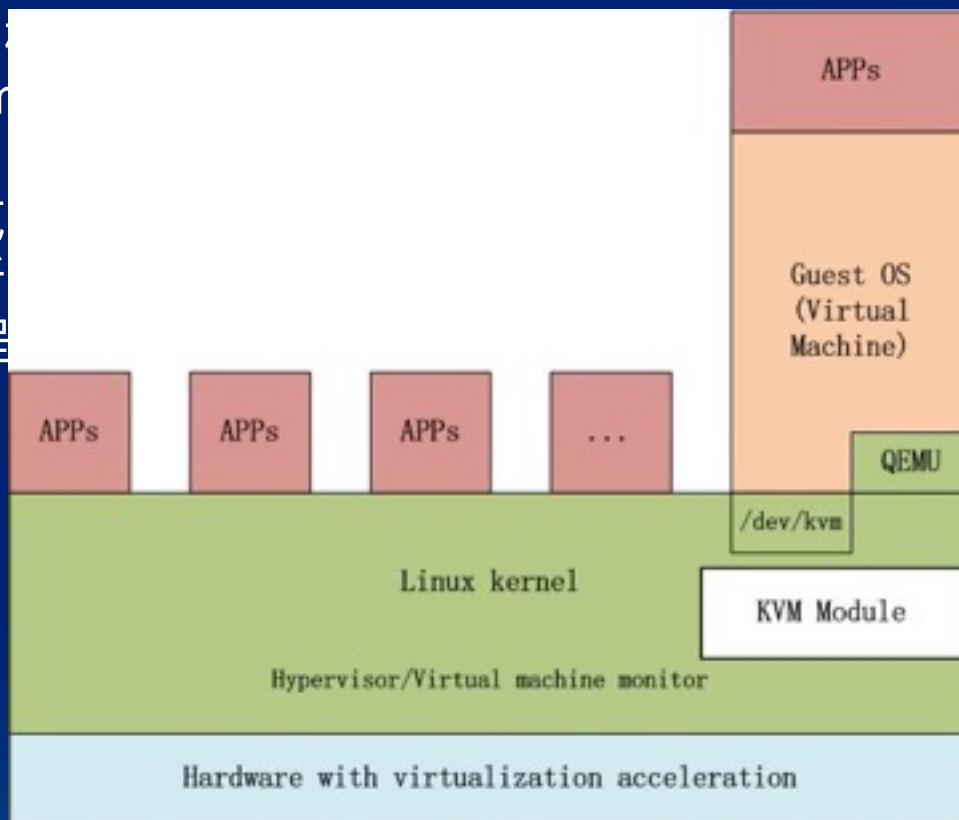
2. KVM网络介绍

3. KVM与其它虚拟化技术对比

4. 性能评测

1.KVM介绍

KVM (Kernel Based Virtual Machine) : KVM 通过加载内核模块而将 Linux 内核导出为一个名为 `/dev/kvm` 的设备。在用户空间，KVM 以用户模式运行，并导出一个名为 `/dev/kvm` 的设备。该设备地址空间独立于内核地址空间，并位于用户空间。在用户空间，KVM 打开 `/dev/kvm` 设备，并导出一个名为 `/dev/kvm` 的设备。



1.KVM介绍

KVM向linux中引入了一种新进程模式--客户模式(guest mode)。

- Kernel mode:表示代码执行的特权模式，通常是内核代码；
- User mode:表示非特权模式，用于执行在内核之外的程序；
- Guest mode: 用来执行客户操作系统代码，但只针对那些非I/O代码。客户模式有自己的内核和用户模式，因此客户操作系统在客户模式中运行可以支持标准的内核，而在用户模式下运行则支持自己的内核和用户 空间应用程序。

1.KVM介绍

KVM 只是虚拟化解决方案的一部分。

- 处理器（目前指 Intel VT 或 AMD-SVM 处理器）直接提供了虚拟化支持。
- KVM 通过 /dev/kvm 设备提供了内存虚拟化。每个客户操作系统都有自己的地址空间，并且是在实例化客户操作系统时映射的。映射给客户操作系统的物理内存实际上是映射给这个进程的虚拟内存。
- I/O 通过一个稍加修改的 QEMU 进程（执行每个客户操作系统进程的一个拷贝）进行虚拟化。

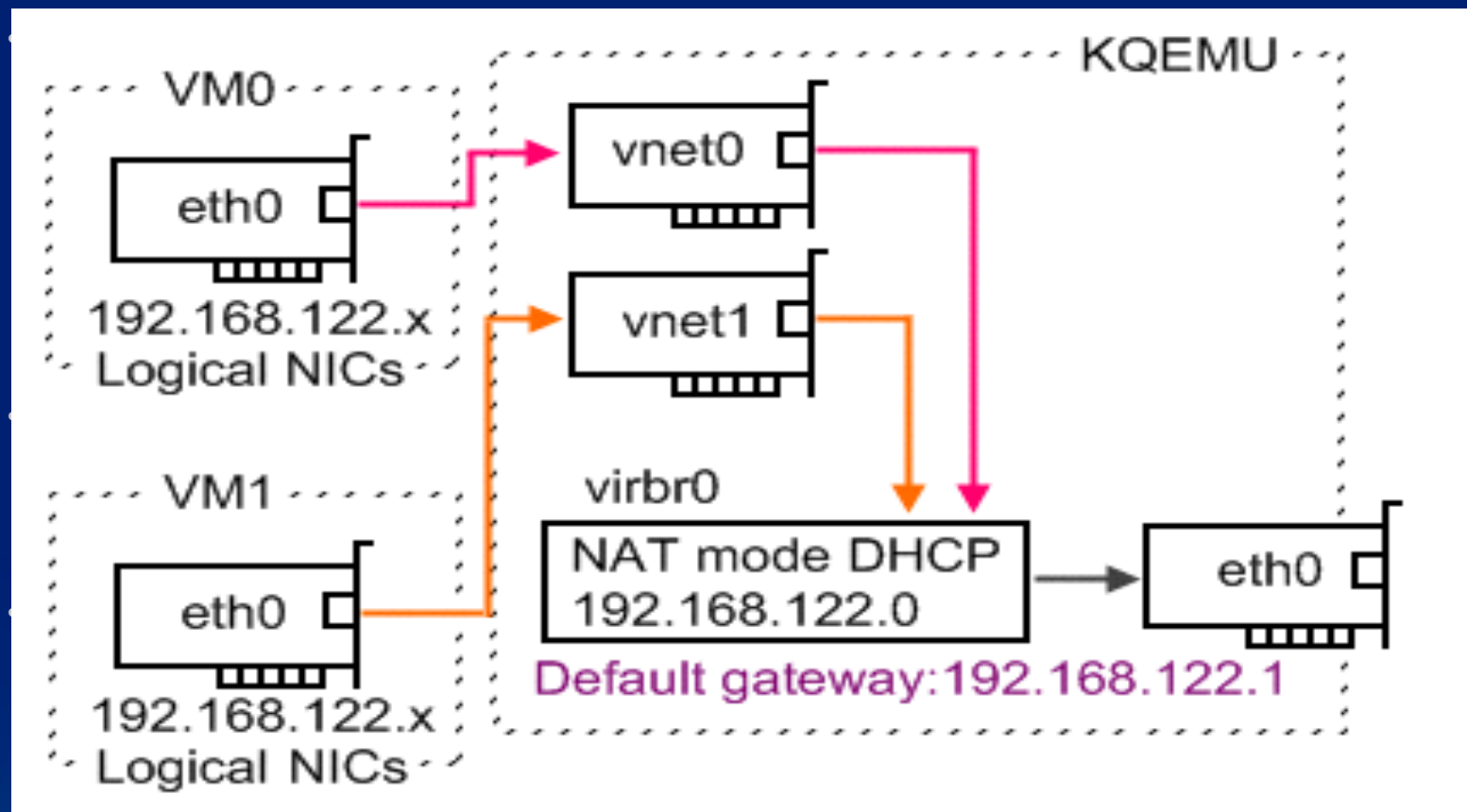
2.KVM网络配置

设置KVM虚拟机时有两种网络选项：

1. Virtual network
2. Shared physical device

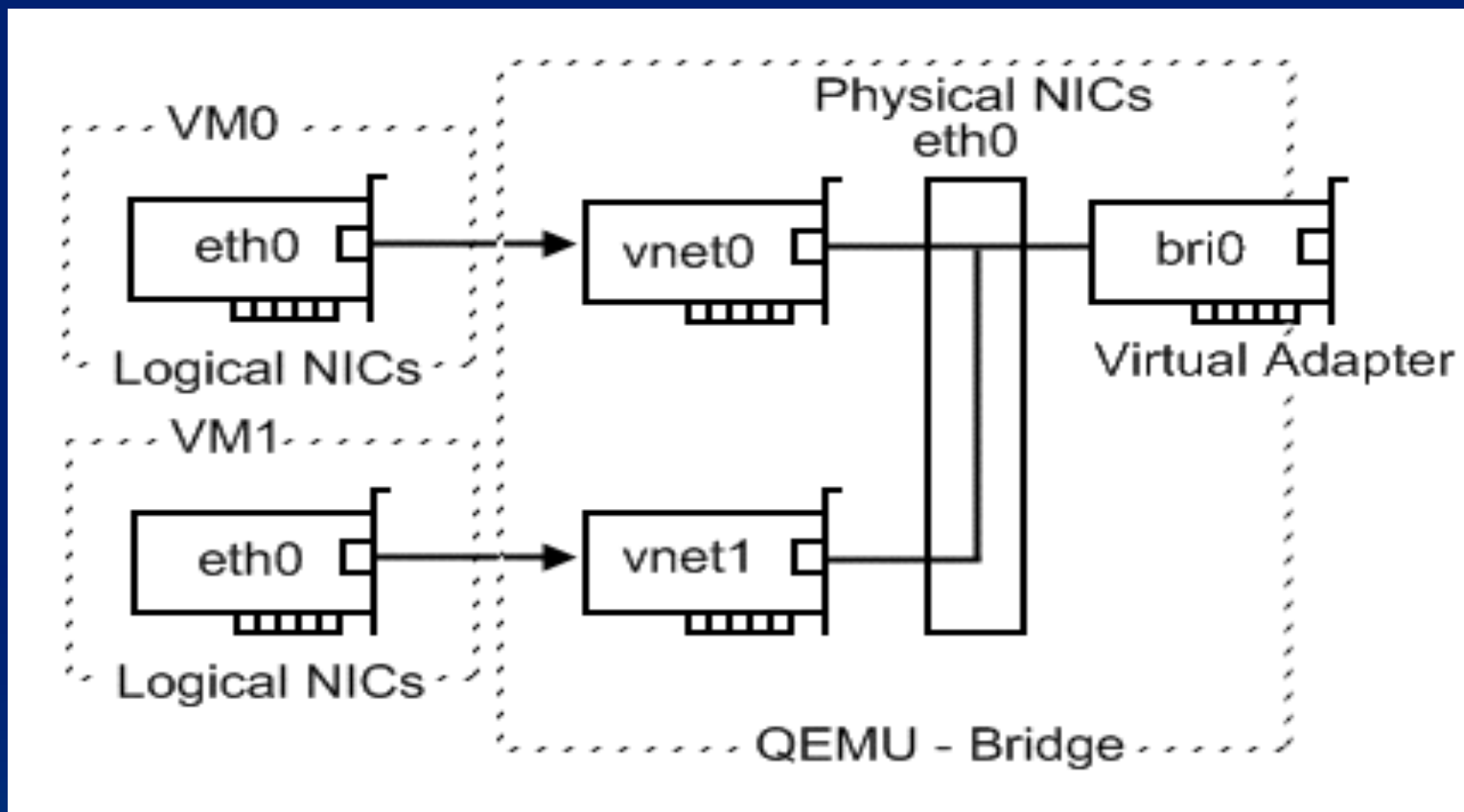
2.KVM网络配置

Virtual network - virbr0



2.KVM网络配置

Shared physical device



3. KVM与其它虚拟化技术对比

与Linux相关的虚拟化项目：

项目	类型	许可证
Bochs	仿真	LGPL
QEMU	仿真	LGPL/GPL
KVM	完全虚拟化	LGPL/GPL
VMware	完全虚拟化	私有
Z/VM	完全虚拟化	私有
Xen	超虚拟化	GPL
UML	超虚拟化	GPL
Linux-VServer	操作系统级虚拟化	GPL
OpenVZ	操作系统级虚拟化	GPL

4.性能评测

- 硬件部分:

处理器: 2 x Intel Xeon LV Dual-Core 2.00GHz

主板: Tyan Tiger i7520SD S5365

内存: 2 x 512MB Mushkin ECC Reg DDR2-533

显卡: NVIDIA GeForce FX5200 128MB PCI

硬盘: Western Digital 160GB SATA2

光驱: Lite-On 16x DVD-ROM

冷却部件: 2 x Dynatron Socket 479 HSFs

机箱: SilverStone Lascala LC20

电源: SilverStone Strider 560W

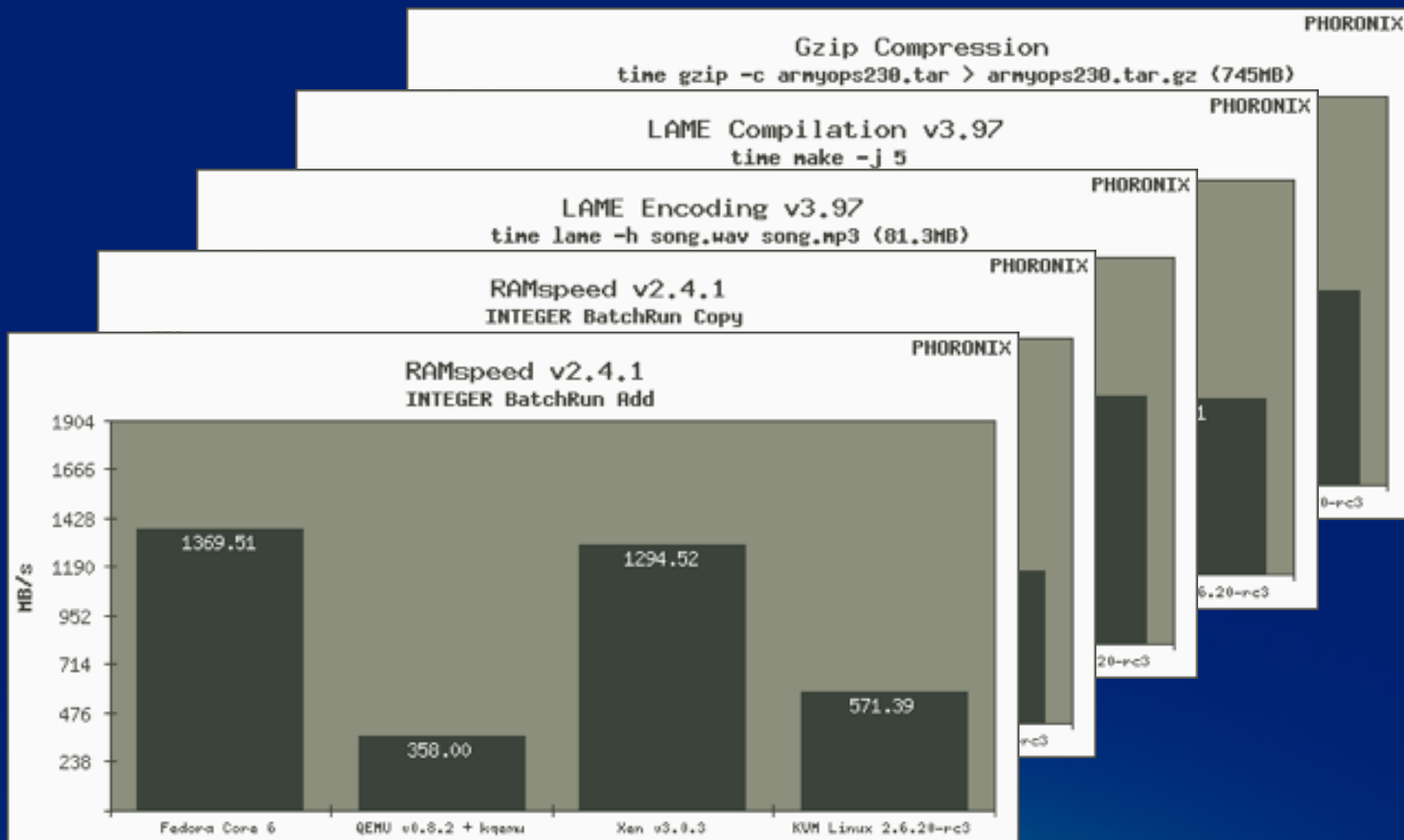
- 软件部分:

操作系统: Fedora Core 6

4.性能评测

选用的基准测试包括Gzip压缩, LAME Complication(编译), LAME encoding(编码)和RAMspeed。我们使用的虚拟环境 QEMU V0.8.2(含kqemu accelerator模块), Xen 3.0.3和KVM。我们也比较了这些虚拟化环境和没有任何虚拟化技术的Fedora Core 6 Zod。在Xen 3.0.3测试当中, 我们使用了全虚拟化而不是半虚拟化技术。测试中使用的映像文件大小为10GB。整个测试过程使用的操作系统是Fedora Core 6 Zod。

4.性能评测



4.性能评测

评测结论：

KVM在Gzip压缩测试中是领先的，但是在其它四个测试中都落后于Xen 3.0.3。然而，所有测试中Xen的全虚拟化和KVM都领先于QEMU+kqemu。KVM的优点是高性能，稳定，无需修改客户机系统和大量的其它功能(比如，可使用linux调度器)。



三、在RHEL5.6上部署安装说明

1.安装RHEL5.6-X64操作系统和相关软件包

2.配置KVM网络

3.安装客户操作系统

1.安装RHEL操作系统和相关软件包

- 检查BIOS，是否开启CPU的VT选项，注意开启后要冷启动(对于SATA硬盘，最好把硬盘运行模式设置为AHCI模式)；
- 安装RHEL5.6操作系统；
 - a.安装完后可以查看内核版本确认是否支持KVM(2.6.18以上版本支持，最好使用最新linux内核)；
 - b.检查CPU配置文件确认CPU是否支持KVM；

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 2.6.18-238.el5 #1 SMP Sun Dec 19 14:22:44 EST 2010 x86_64 x86_64 x86_64 GNU/Linux
```

```
[root@localhost ~]# egrep "(vmx|svm)" /proc/cpuinfo
flags      : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm syscall nx lm constant_tsc ida pni monitor ds_cpl vmx smx est tm2 sse3 c
x16 xtpr sse4_1 lahf_lm
flags      : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm syscall nx lm constant_tsc ida pni monitor ds_cpl vmx smx est tm2 sse3 c
x16 xtpr sse4_1 lahf_lm
```

- 安装KVM相关软件包、启动libvirt服务并检查服务是否启动；

```
[root@localhost ~]# chkconfig libvirtd on x86_64 virt-*
```

2.配置KVM网络

- 停止物理网卡eth0
- 修改eth0的网卡配置 `[root@localhost ~]# ifdown eth0` `scripts/ifcfg-eth0`

- 创建一个新的网络接口br0

- 启动网络

(注意：如
照下面的

- 确定设置
务

- 确保网关已设置

- 确保可用

再测试，应

```
[root@localhost network-scripts]# cat /etc/sysconfig/network-scripts/ifcfg-eth0
# [root@localhost network-scripts]# cat /etc/sysconfig/network-scripts/ifcfg-br0
DEVICE=br0
BOOTPROTO=static
ONBOOT=yes
IPADDR=192.168.190.254
NETMASK=255.255.255.0
GATEWAY=192.168.190.254
ONBOOT=yes
```

```
[root@localhost ~]# cat /etc/resolv.conf
```

```
[root@localhost ~]# ping www.baidu.com
PING www.a.shifen.com (61.135.169.125) 56(84) bytes of data:
64 bytes from 61.135.169.125: icmp_seq=1 ttl=53 time=8.47 ms
64 bytes from 61.135.169.125: icmp_seq=2 ttl=53 time=8.71 ms
64 bytes from 61.135.169.125: icmp_seq=3 ttl=53 time=5.44 ms
```

```
--- www.a.shifen.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 5.440/7.543/8.712/1.490 ms
```

```
ts/ifcfg-*
254
```

3.安装客户操作系统

(注意：如果安装上面步骤进行设置后还是不能连接外网，可以按照下面的步骤进行检查：

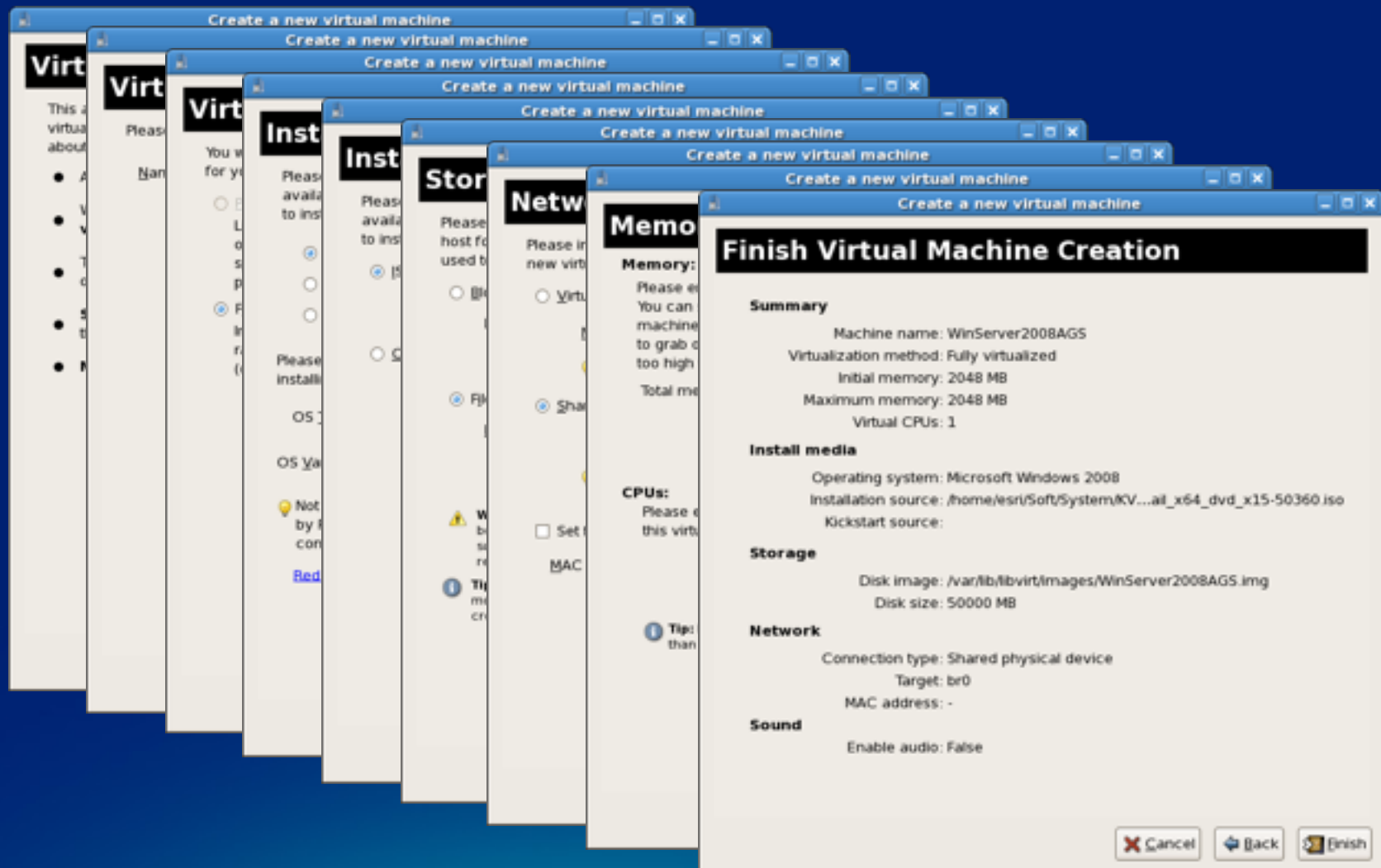
- 确定设置了域名服务器, 没有的话, 建议设置Google的公共DNS服务
- 确保网关已设置
- 确保可用dns解析

再测试，应该就没什么问题了。)

3.安装客户操作系统

- 停止物理网卡eth0
 - 修改eth0的网卡配置/etc/sysconfig/network-scripts/ifcfg-eth0
 - 创建一个新的网络接口br0
 - 启动网络eth0和br0
(注意：如果安装上面步骤进行设置后还是不能连接外网，可以按照下面的步骤进行检查：
 - 确定设置了域名服务器，没有的话，建议设置Google的公共DNS服务
 - 确保网关已设置
 - 确保可用dns解析
- 再测试，应该就没什么问题了。)

3.安装操作系统



谢谢!

