# 云计算技术与工程

## 虚拟化技术

徐敏贤

副研究员
云计算研究中心
中国科学院深圳先进技术研究院
http://www.minxianxu.info/vcc

Group: 2024秋《云计算技术与工程》

浮云在虚空，随风复卷舒。
——（唐）储光羲

# 内容回顾

- Q1: 请解释为什么会出现云计算?

- ✓ 互联网上的**数据量**高速增长，导致了互联网数据处理能力不足；
- ✓ 互联网上存在着大量处于**闲置状态**的计算设备和存储资源；
- ✓ 服务器更新换代速度加快，企业升级**费用昂贵**。

# 内容回顾

- Q2: 给出一种云计算的定义?

- Definition from ***NIST*** ***(National Institute of Standards and Technology)***
  - Cloud computing is a model for enabling convenient, **on-demand network access** to a **shared pool** of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be **rapidly provisioned and released** with minimal management effort or service provider interaction.
  - This cloud model promotes **availability** and is composed of five essential characteristics, three service models, and four deployment models.

# 内容回顾

- Q3: 云计算有哪些服务模式?

基础设施即服务Infrastructure as a Service (IaaS)
平台即服务Platform as a Service (PaaS)
软件即服务Software as a Service (SaaS)

# 内容回顾

- Q4: 云计算的目标和特点?

- 目标：像用电、水一样使用IT
- 特点：

# 目录 Contents



注：本章部分内容参考刘鹏、王洋教授

# 1 虚拟化技术简介

# 虚拟化技术简介

- 虚拟化技术是伴随着计算机的出现而产生和发展起来的，虚拟化意味着对计算机资源的抽象。
- 虚拟化技术已经成为构建云计算环境的一项关键技术。

# 20世纪60年代

## IBM公司推出虚拟化技术

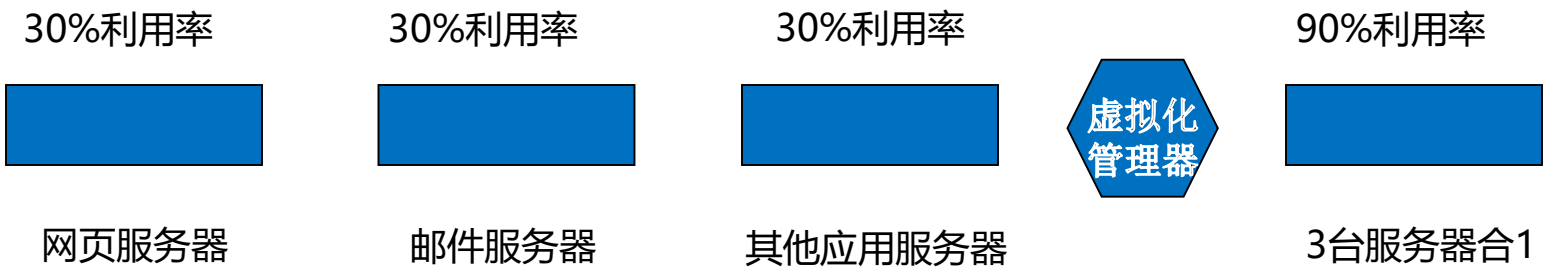
主要用于当时的IBM大型机的服务器虚拟化

---

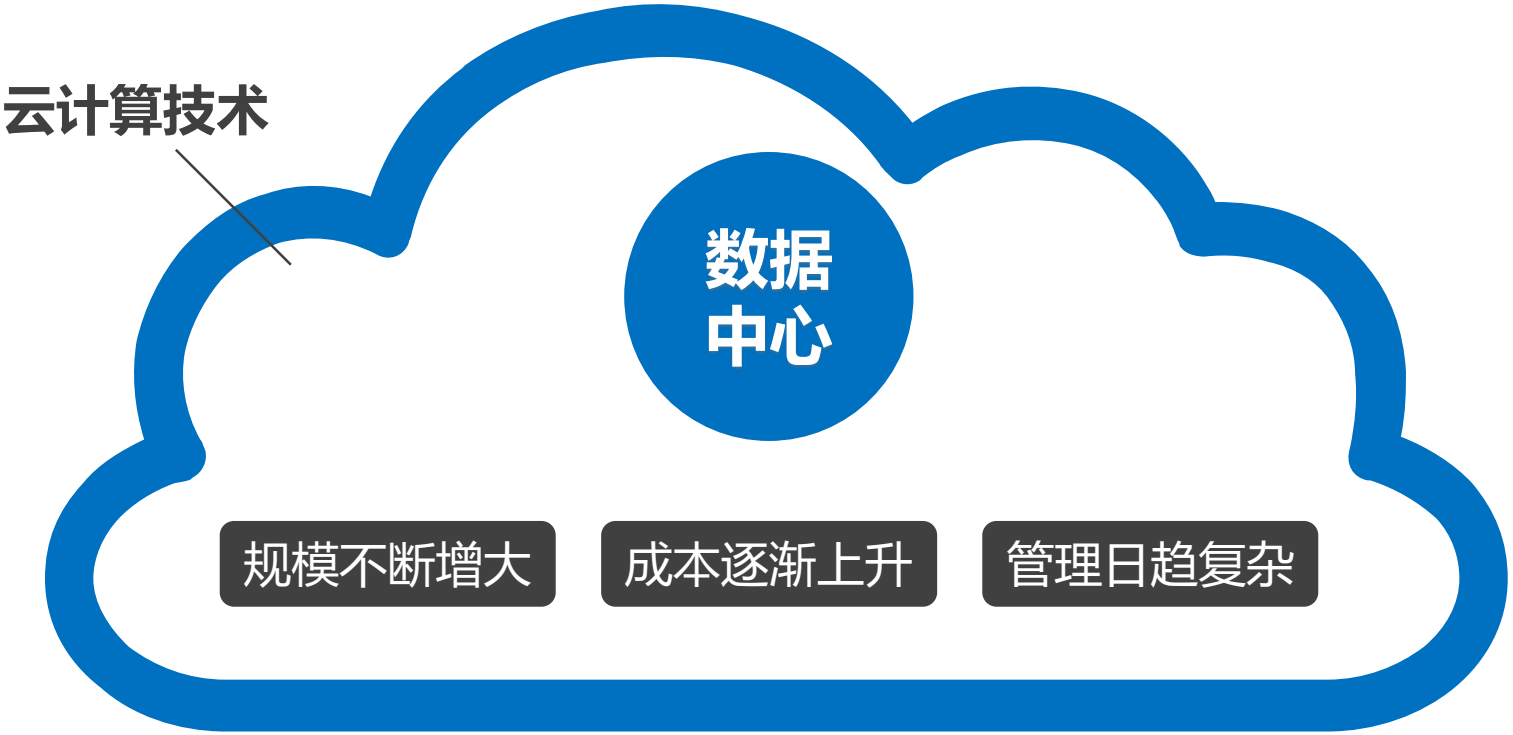
虚拟化技术的核心思想是利用软件或固件管理程序构成虚拟化层，把物理资源映射为虚拟资源。在虚拟资源上可以安装和部署多个虚拟机，实现多用户共享物理资源。

# 虚拟化的工作原理

- 例子：


30%利用率
30%利用率
30%利用率
虚拟化
管理器
90%利用率
网页服务器
邮件服务器
其他应用服务器
3台服务器合1

# 虚拟化技术简介



# 与传统数据中心对比

**传统的数据中心**

| 采用了多种技术 | 业务之间孤立 | 网络结构复杂 |
| --- | --- | --- |

**虚拟数据中心**

| 高速 | 扁平 | 虚拟化 |
| --- | --- | --- |

# 虚拟化技术的优点

随着云计算的发展，传统的数据中心逐渐过渡到虚拟化数据中心，即采用虚拟化技术将原来数据中心的物理资源进行抽象整合。

- 实现资源的动态分配和调度，提高现有资源的利用率和服务可靠性
- 提供自动化的服务开通能力，降低运维成本
- 具有有效的安全机制和可靠性机制，满足公众客户和企业客户的安全需求
- 方便系统升级、迁移和改造

# 虚拟化的类别

**数据中心的虚拟化**

**服务器虚拟化**

将一个或多个物理服务器虚拟成多个逻辑上的服务器

**存储虚拟化**

把分布的异构存储设备统一为一个或几个大的存储池

**网络虚拟化**

在底层物理网络和网络用户之间增加一个抽象层

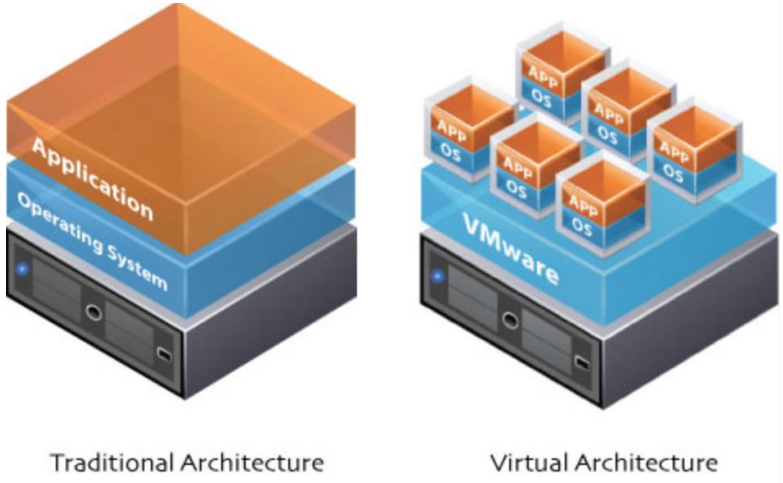
# 安装虚拟机就像.....

# 2

# 服务器虚拟化

# 服务器虚拟化

- 一个物理的服务器虚拟成若干个独立的逻辑服务器，比如分区
- 把若干分散的物理服务器虚拟为一个大的逻辑服务器，比如网格技术




Ref: snel.com

# 服务器虚拟化的层次

## 寄居虚拟化

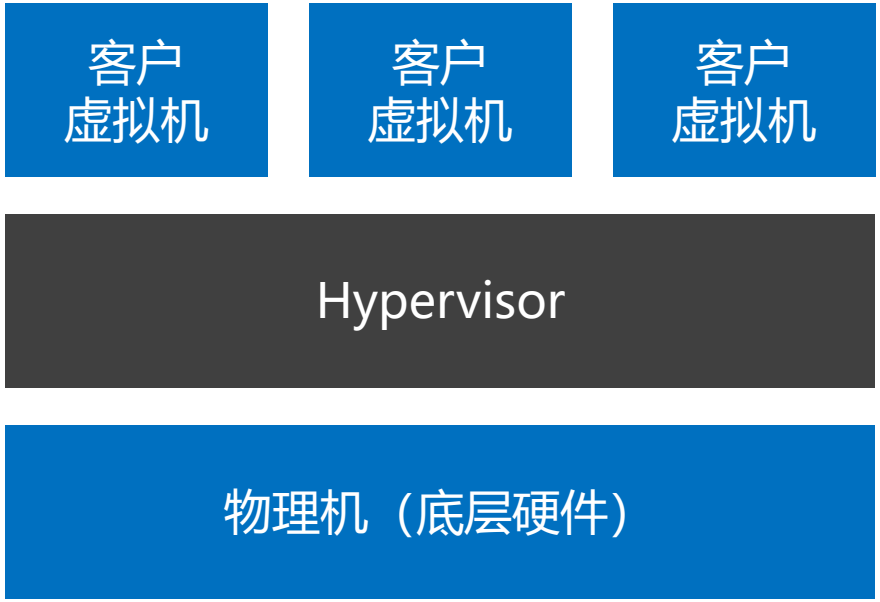
- 寄居虚拟化的虚拟化层一般称为虚拟机监控器（VMM）
- 这类虚拟化架构系统损耗比较大
- 就操作系统层的虚拟化而言，没有独立的Hypervisor层
- 如果使用操作系统层虚拟化，所有虚拟服务器必须运行同一操作系统


寄居虚拟化架构

# 服务器虚拟化的层次

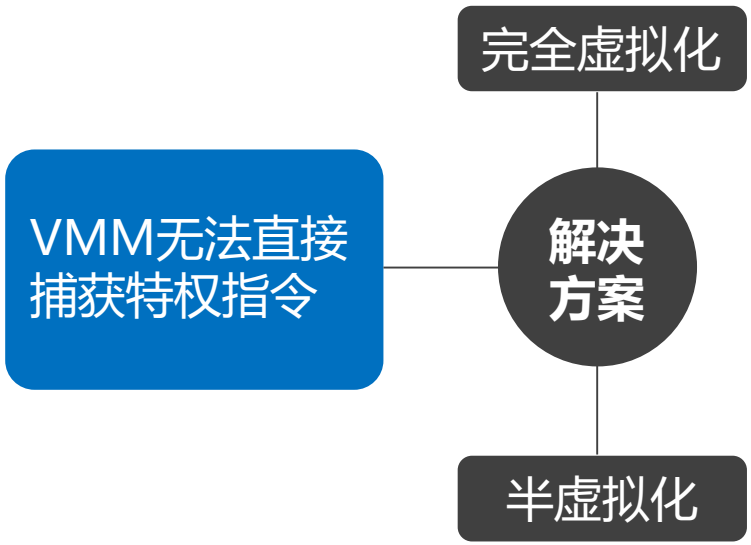
## 裸机(Bare Metal)虚拟化

- 架构中的VMM也可以认为是一个操作系统，一般称为Hypervisor
- Hypervisor实现从虚拟资源到物理资源的映射
- Hypervisor实现了不同虚拟机的运行上下文保护与切换，保证了各个客户虚拟系统的有效隔离




裸机虚拟化架构

# 服务器虚拟化的层次




完全虚拟化具有很好的兼容性，在服务器虚拟化中得到广泛应用。

半虚拟化技术降低了由于虚拟化而引入的系统性能损失。

# 主流的VMM（Hypervisor）

- **KVM** – a Linux based open source hypervisor. First introduced into the Linux kernel in February 2007, it is now a mature hypervisor and is probably the most widely deployed open source hypervisor in an open source environment. KVM is used in products such as Redhat Enterprise Virtualization (RHEV).


- **Xen** – An open source hypervisor which originated in a 2003 Cambridge University research project. It runs on Linux (though being a Type 1 hypervisor, more properly one might say that its dom0 host runs on Linux, which in turn runs on Xen). It was originally supported by XenSource Inc, which was acquired by Citrix Inc in 2007.

# 主流的VMM（Hypervisor）

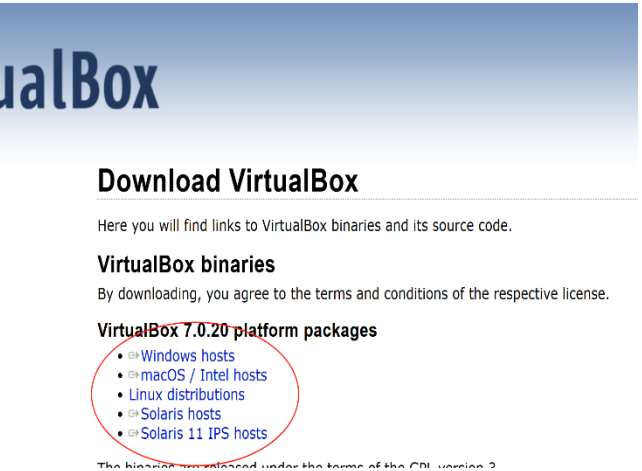
- **VMware** – is not a hypervisor, but the name of a company, VMware Inc. VMware involves hypervisor in its vSphere product. vSphere uses VMware's ESXi hypervisor. VMware's hypervisor is very mature and extremely stable.

- **Hyper-V** – Hyper-V is a commercial hypervisor provided by Microsoft. Whilst excellent for running Windows, being a hypervisor it will run any operating system supported by the hardware platform.

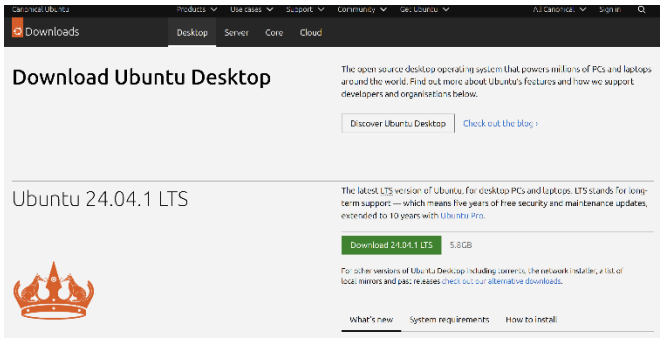
- **VirtualBox** - VirtualBox is a powerful x86 and AMD64/Intel64 virtualization product for enterprise as well as home use.

# Demo with VirtualBox

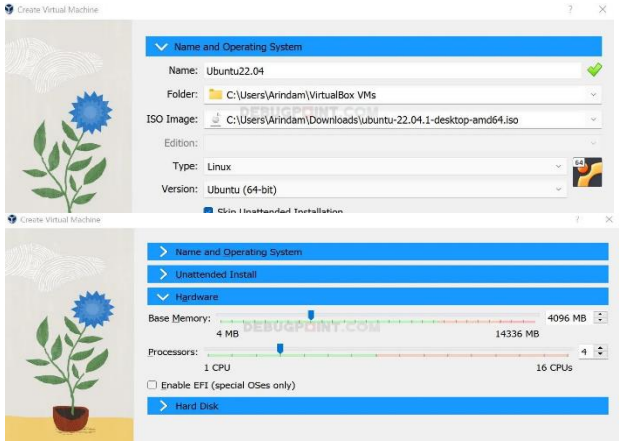

https://www.virtualbox.org/




Download Suitable Version

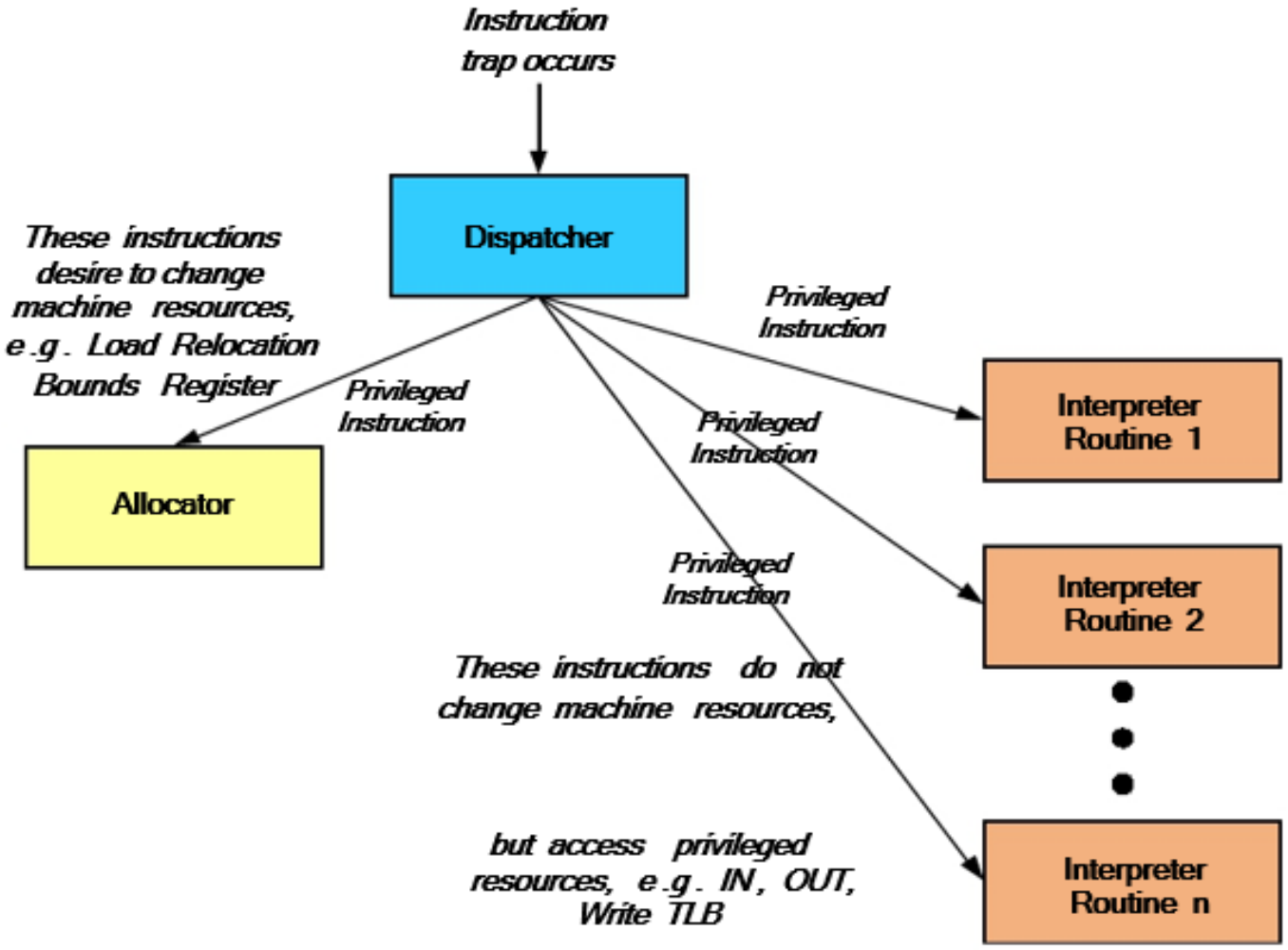



https://ubuntu.com/download/desktop




安装教程：https://linux.cn/article-15472-1.html

# VMM组件


Instruction
trap occurs
Dispatcher
These instructions
desire to change
machine resources,
e.g. Load Relocation
Bounds Register
Privileged
Instruction
Allocator
Privileged
Instruction
Privileged
Instruction
Interpreter
Routine 1
Privileged
Instruction
Interpreter
Routine 2
These instructions do not
change machine resources,
but access privileged
resources, e.g. IN, OUT,
Write TLB
Interpreter
Routine n

# VMM组件

- Dispatcher
  - control module, called on hardware traps
  - decides the next module to be invoked
- Allocator
  - decides allocation of system resources
  - invoked by dispatcher to change machine resource allocation for VMs
- Interpreter routines
  - emulates trap functionality for each user VM
  - all traps except resource re-assignment traps

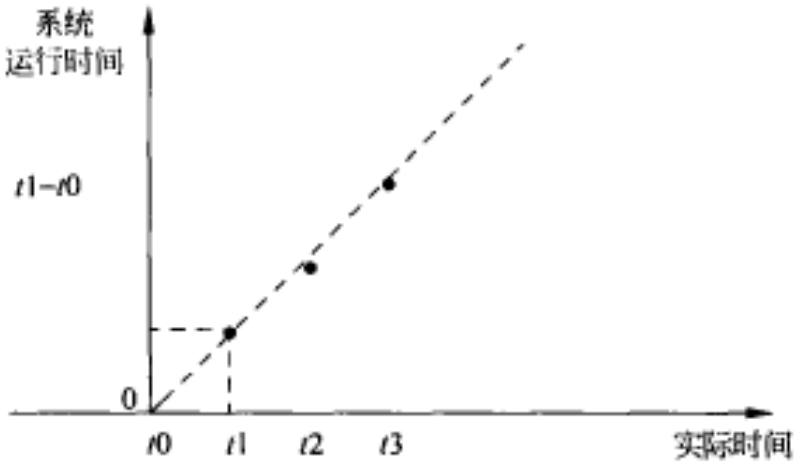
# VMM特性

- Efficiency
  - All *innocuous (safe)* instructions *must* be natively executed
- Resource control
  - impossible for guest s/w to *directly* change system resource configurations
- Equivalence
  - guaranty of identical behavior
  - allowed exceptions
    - performance can be slower
    - available resources can be limited
    - differences in performance due to changed timings

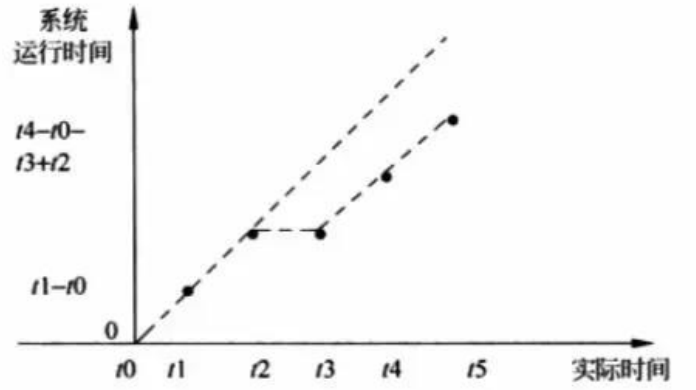
# VMM辅助

- **操作系统的时间概念**
  - 绝对时间(Wall Time)
  - 相对时间
  - 当前绝对时间=系统启动时间+系统启动后运行时间

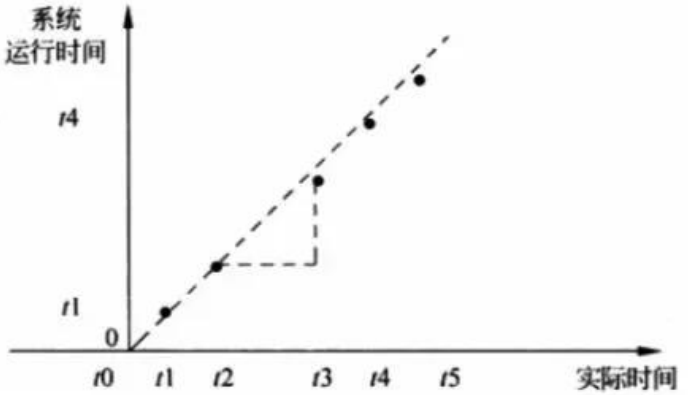
  举例：t0启动，运行到t1




- **客户机的时间概念**



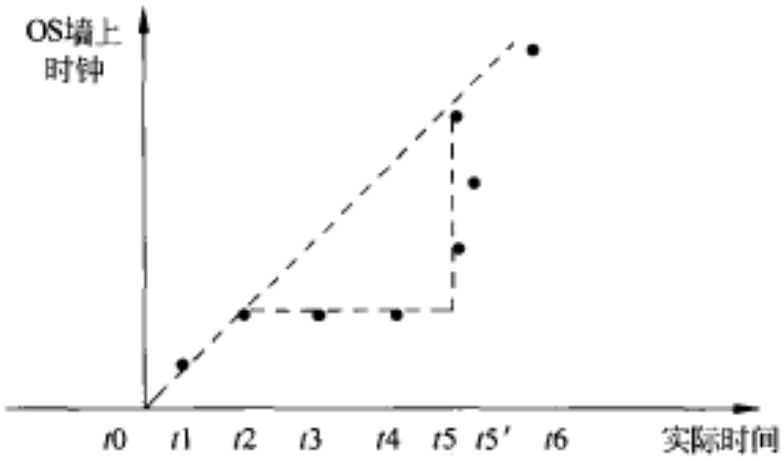

客户机的时间概念(客户机时间！＝实际时间)

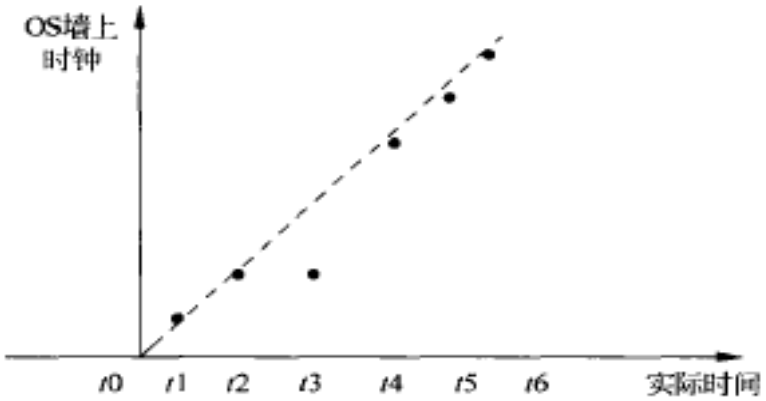



客户机时间概念(客户机时间＝＝实际时间)

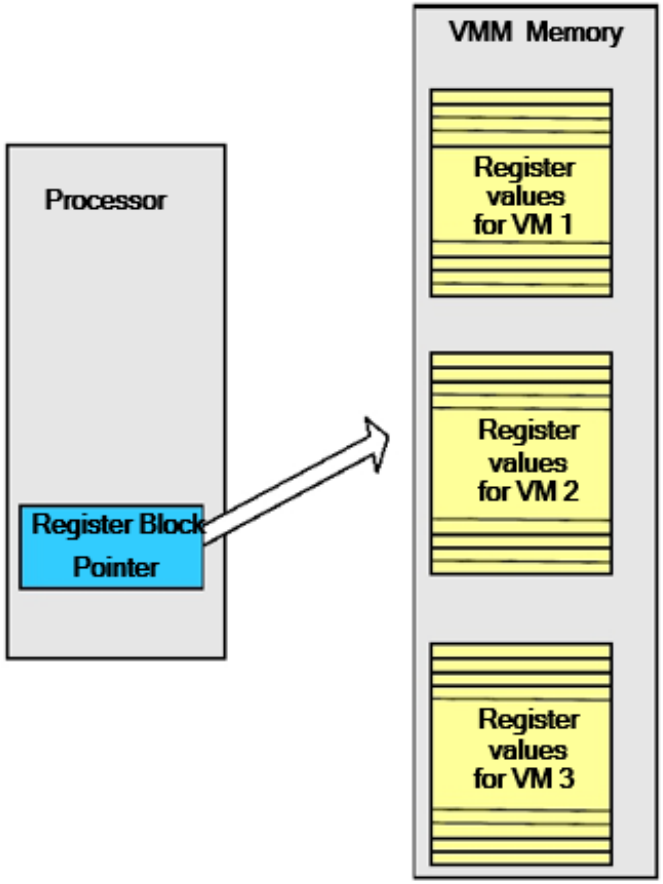
# 时间虚拟化

- **客户机时间修正**
  - 客户机时间=客户机实际运行时间 (进程记账)
  - **客户机时间= 实际时间**（网络速度检测程序）
- **实现客户机时间概念方法**
  - 丢失中断补偿法
  - 时钟设备计数器法



# 状态管理



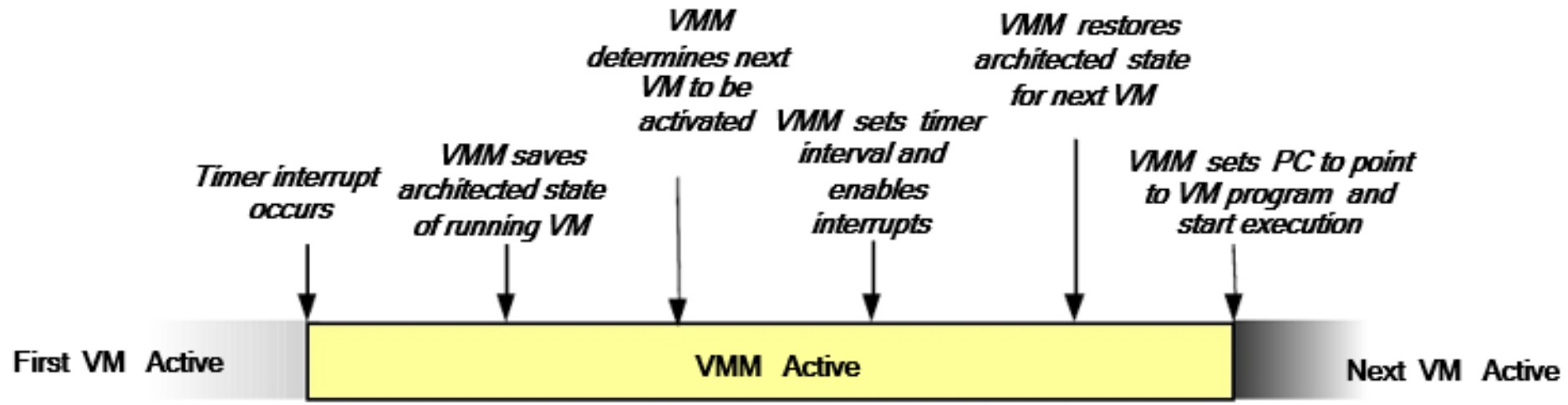

```
Load register block pointer
to point to VM's registers
in VMM memory

Load program counter to
point to VM program and
start execution
                •
                •
                •
load temp <- reg_pointer, index(A)
store reg_pointer, index(B) <- temp

                •
                •
                •
```

# 资源控制

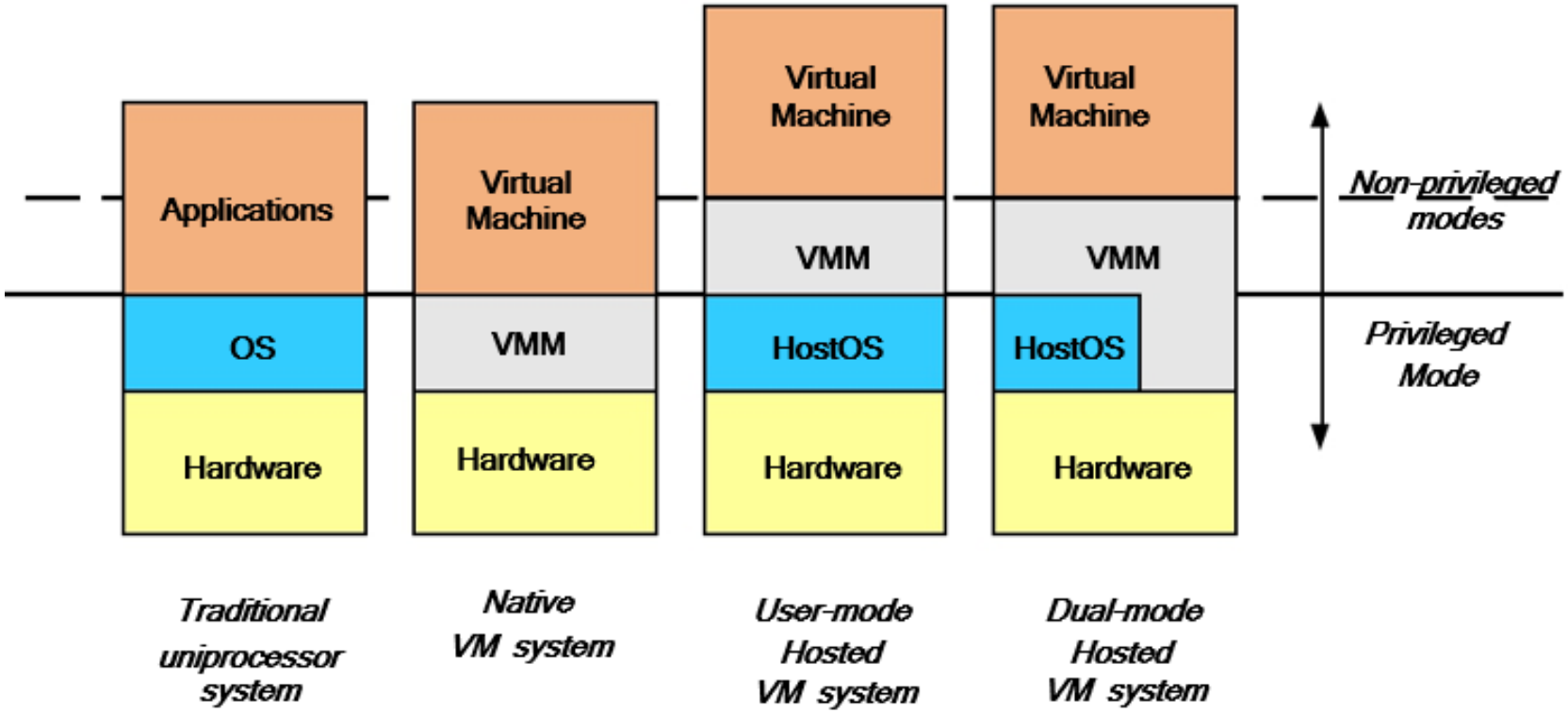
- VMM对应用资源实现全方位控制
- 类似操作系统的时分功能
  - VMM能够实现资源抢占
  - VMM能够实现特权指令控制
  - VMM能够处理时钟中断
  - 平衡公平调度和巨大的切换开销

# VMM的时分操作

- 实现在不同用户之间的资源时分
  - 类似于操作系统的时分



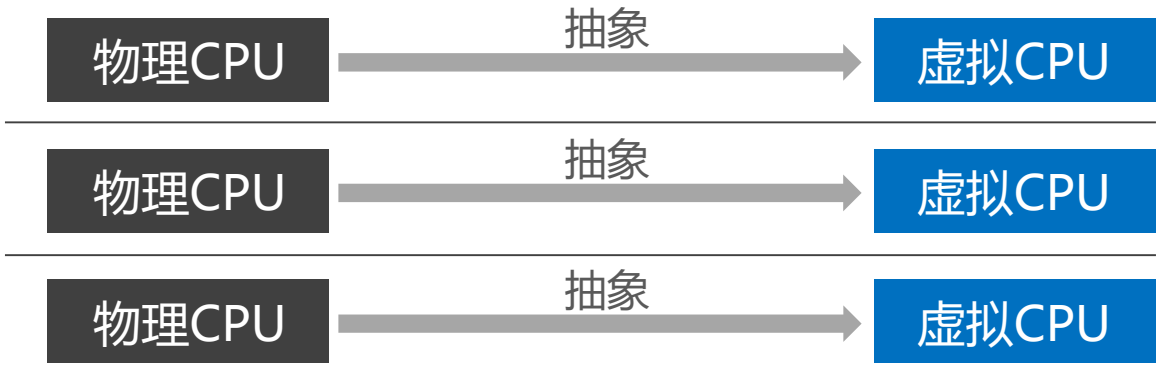
# 原生与寄宿虚拟机


Applications
OS
Hardware
Traditional uniprocessor system
Virtual Machine
VMM
Hardware
Native VM system
Virtual Machine
VMM
HostOS
Hardware
User-mode Hosted VM system
Virtual Machine
VMM
HostOS
Hardware
Dual-mode Hosted VM system
Non-privileged modes
Privileged Mode

# 服务器虚拟化的底层实现

## CPU虚拟化

客户
操作系统




1. 虚拟CPU的正确运行是要保证虚拟机指令正确运行，现有的实现技术包括模拟执行和监控执行

2. 调度问题是指VMM决定当前哪个虚拟CPU在物理CPU上运行，要保证隔离性、公平性和性能。

# 服务器虚拟化的底层实现

## 内存虚拟化

内存虚拟化技术把物理内存统一管理，包装成多个虚拟的物理内存提供给若干虚拟机使用，每个虚拟机拥有各自独立的内存空间。

虚拟内存的管理包括3种地址

# 服务器虚拟化的底层实现

## I/O设备虚拟化

I/O设备虚拟化技术把真实的设备统一管理起来，包装成多个虚拟设备给若干个虚拟机使用，响应每个虚拟机的设备访问请求和I/O请求。

I/O设备虚拟化同样是由VMM进行管理的

- 全虚拟化
- 半虚拟化
- 软件模拟

# 虚拟机迁移

虚拟机迁移是将虚拟机实例从源宿主机迁移到目标宿主机，并且在目标宿主机上能够将虚拟机运行状态恢复到其在迁移之前相同的状态，以便能够继续完成应用程序的任务。

从虚拟机迁移的源与目的地角度可分为

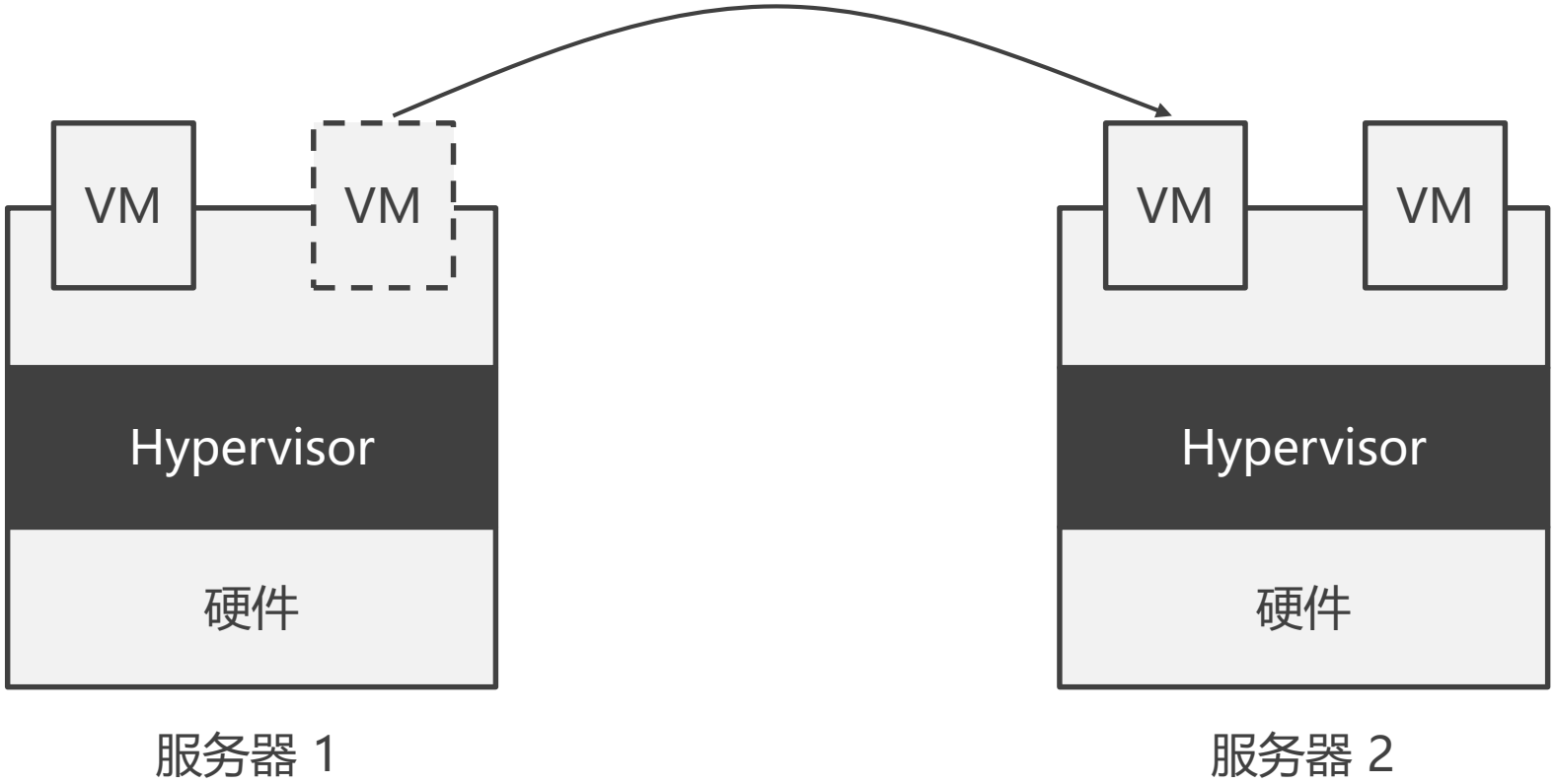
- 物理机到虚拟机的迁移（P2V）
- 虚拟机到虚拟机的迁移（V2V）
- 虚拟机到物理机的迁移（V2P）

# 虚拟机迁移

实时迁移（LiveMigration），就是保持虚拟机运行的同时，把它从一个计算机迁移到另一个计算机，并在目的计算机恢复运行的技术。

| | |
|---|---|
| 第一 | 云计算中心的物理服务器负载经常处于动态变化中，当一台物理服务器负载过大时，若此刻不可能提供额外的物理服务器，管理员可以将其上面的虚拟机迁移到其他服务器，达到负载平衡 |
| 第二 | 云计算中心的物理服务器有时候需要定期进行升级维护，当升级维护服务器时，管理员可以将其上面的虚拟机迁移到其他服务器，等升级维护完成之后，再把虚拟机迁移回来 |

# 虚拟机迁移例子




虚拟机迁移示意图

# 虚拟机迁移步骤

**步骤1**

预迁移

(Pre-Migration)

**步骤2**

预定资源

(Reservation)

**步骤3**

预复制

(InterativePre-Copy)

**步骤4**

停机复制

(Stop-and-Copy)

**步骤5**

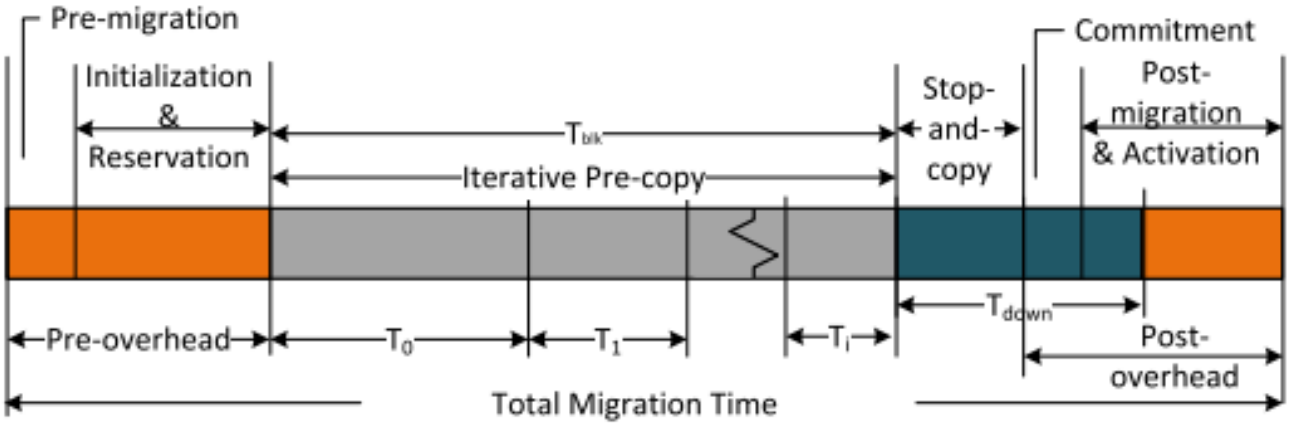
提交

(Commitment)

**步骤6**

启动

(Activation)



# 虚拟机迁移步骤(内存)

内存的迁移是虚拟机迁移最困难的部分

第一阶段，Push阶段。

第二阶段，Stop-and-Copy阶段。

第三阶段，Pull阶段。

实际上，迁移内存没有必要同时包含上述三个阶段，目前大部分的迁移策略只包含其中的一个或者两个阶段。

# 虚拟机迁移方案对比

迁移需要考虑的性能指标：迁移时间，宕机时间，迁移的数据量

| 迁移方案 | 优势与劣势 |
| --- | --- |
| Stop-and-Copy | ● 方法比较简单<br>● 总迁移时间也最短<br>● 停机时间无法接受 |
| Stop-and-Copy和Pull阶段结合 | ● 停机时间很短<br>● 总迁移时间很长<br>● Pull阶段复制造成的性能下降 |
| Push和Stop-and-Copy阶段结合 | ● 平衡了停机时间和总迁移时间之间的矛盾<br>● 需要有一种算法能够测定工作集，以避免反复重传<br>● 可能会占用大量的网络带宽，对其他服务造成影响 |

# 虚拟机迁移步骤(网络)

## 网络资源的迁移

虚拟机这种系统级别的封装方式意味着迁移时VM的所有网络设备，包括协议状态（如TCP连接状态）以及IP地址都要随之一起迁移。

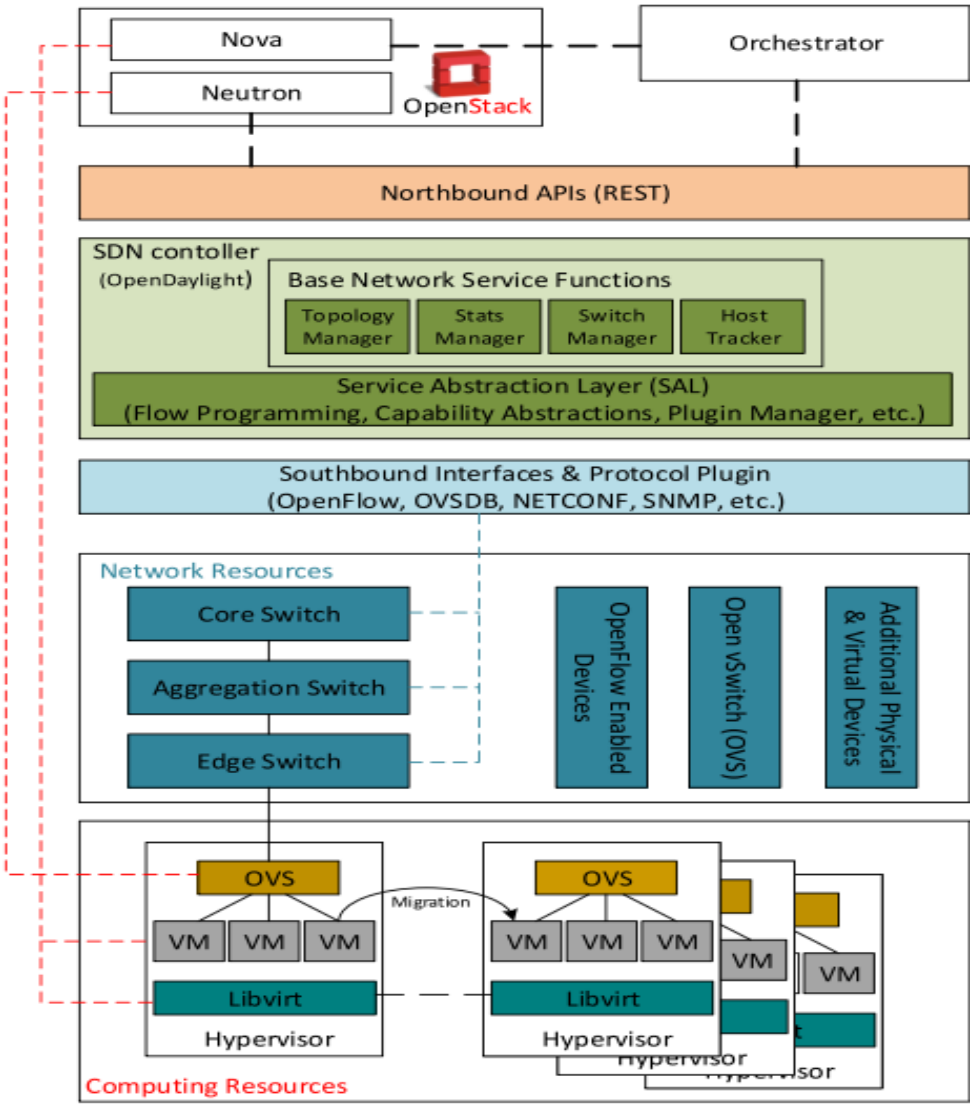
在局域网内，可以通过发送ARP重定向包，将VM的IP地址与目的机器的MAC地址相绑定，之后的所有包就可以发送到目的机器上。

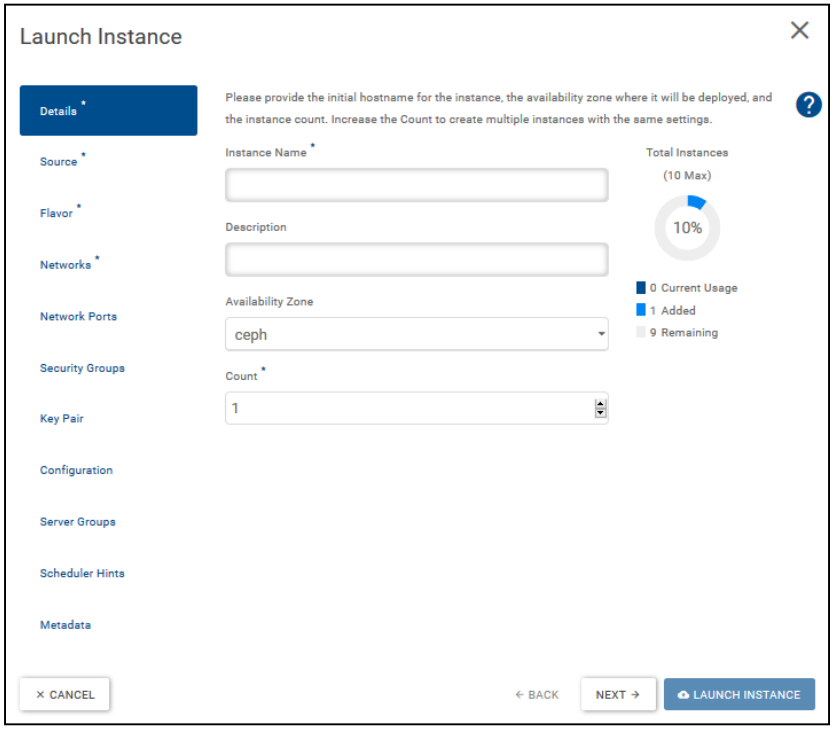
# 虚拟机迁移步骤(存储)

## 存储设备的迁移

- 迁移存储设备的最大障碍在于需要占用大量时间和网络带宽，通常的解决办法是以共享的方式共享数据和文件系统，而非真正迁移。
- 目前大多数集群使用NAS（Network Attached Storage，网络连接存储）作为存储设备共享数据。
- NAS实际上是一个带有瘦服务器的存储设备，其作用类似于一个专用的文件服务器。
- 在局域网环境下，NAS已经完全可以实现异构平台之间，如NT、UNIX等的数据级共享。
- 基于以上的考虑，Xen并没有实现存储设备的迁移，实时迁移的对象必须共享文件系统。

# 基于OpenStack的迁移框架

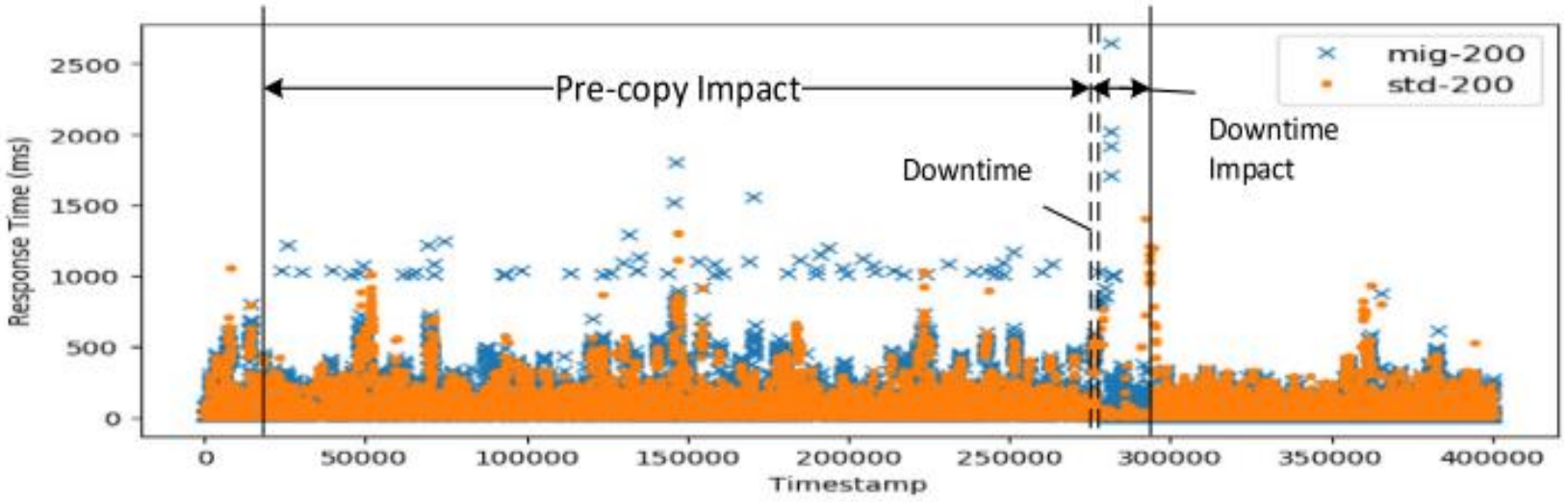



OpenStack创建虚拟机实例

# 迁移过程中对应用性能的影响

- 迁移过程中响应时间显著增加
- Downtime后，出现最差的响应时间，因为需要处理额外的请求



# 隔离技术

虚拟机隔离是指虚拟机之间在没有授权许可的情况下，互相之间不可通信、不可联系的一种技术。

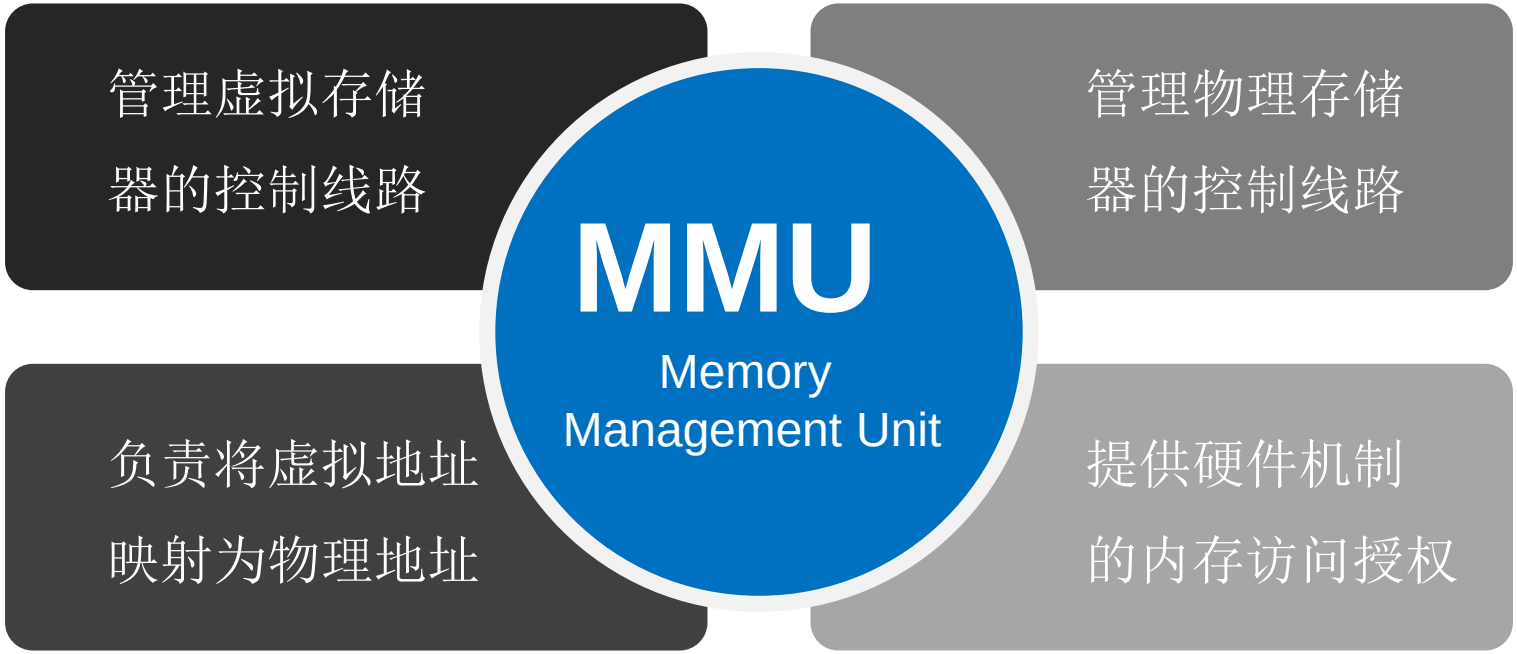
| | |
|---|---|
| 软件角度 | 互相隔离的虚拟机之间保持独立，如同一个完整的计算机 |
| 硬件角度 | 被隔离的虚拟机相当于一台物理机，有自己的CPU、内存、硬盘、I/O等，它与宿主机之间保持互相独立的状态 |
| 网络角度 | 被隔离的虚拟机如同物理机一样，既可以对外提供网络服务，也一样可以从外界接受网络服务 |

# 隔离技术

## 虚拟机隔离机制

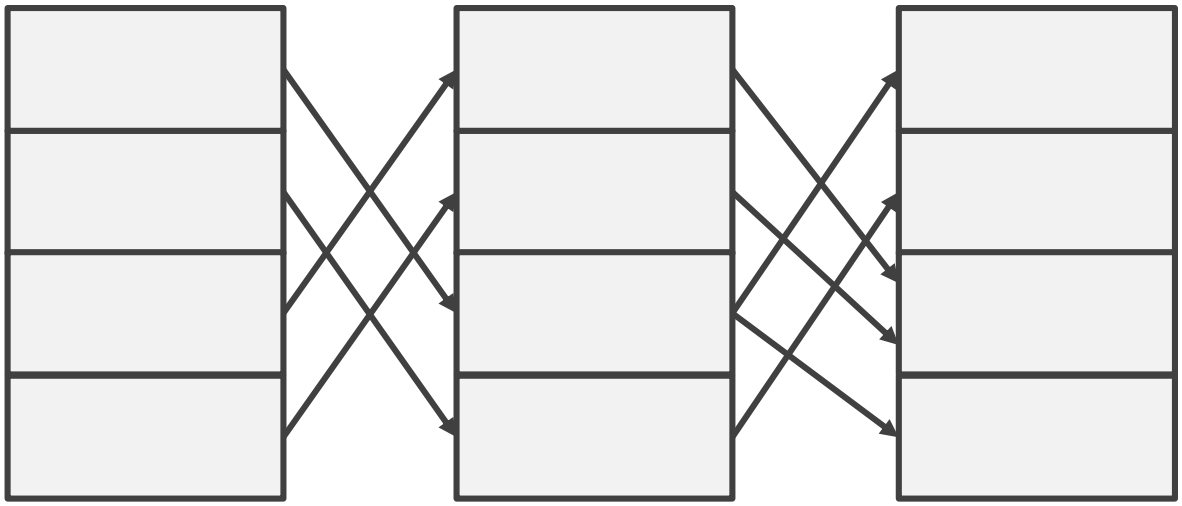
- 网络隔离；
- 构建虚拟机安全文件防护网；
- 基于访问控制的逻辑隔离机制；
- 通过硬件虚拟，让每个虚拟机无法突破虚拟机管理器给出的资源限制；
- 硬件提供的内存保护机制；
- 进程地址空间的保护机制，IP地址隔离。

# 内存隔离

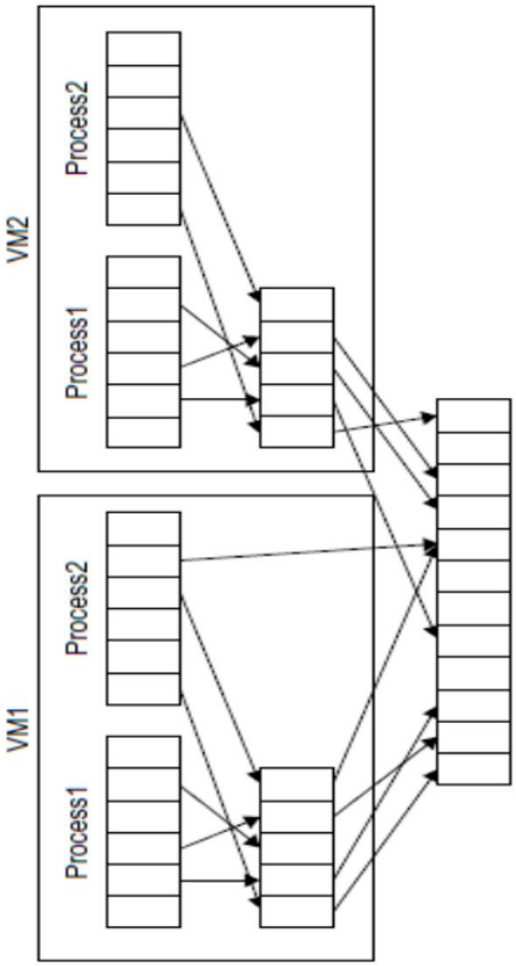

管理虚拟存储器的控制线路
管理物理存储器的控制线路
MMU
Memory Management Unit
负责将虚拟地址映射为物理地址
提供硬件机制的内存访问授权

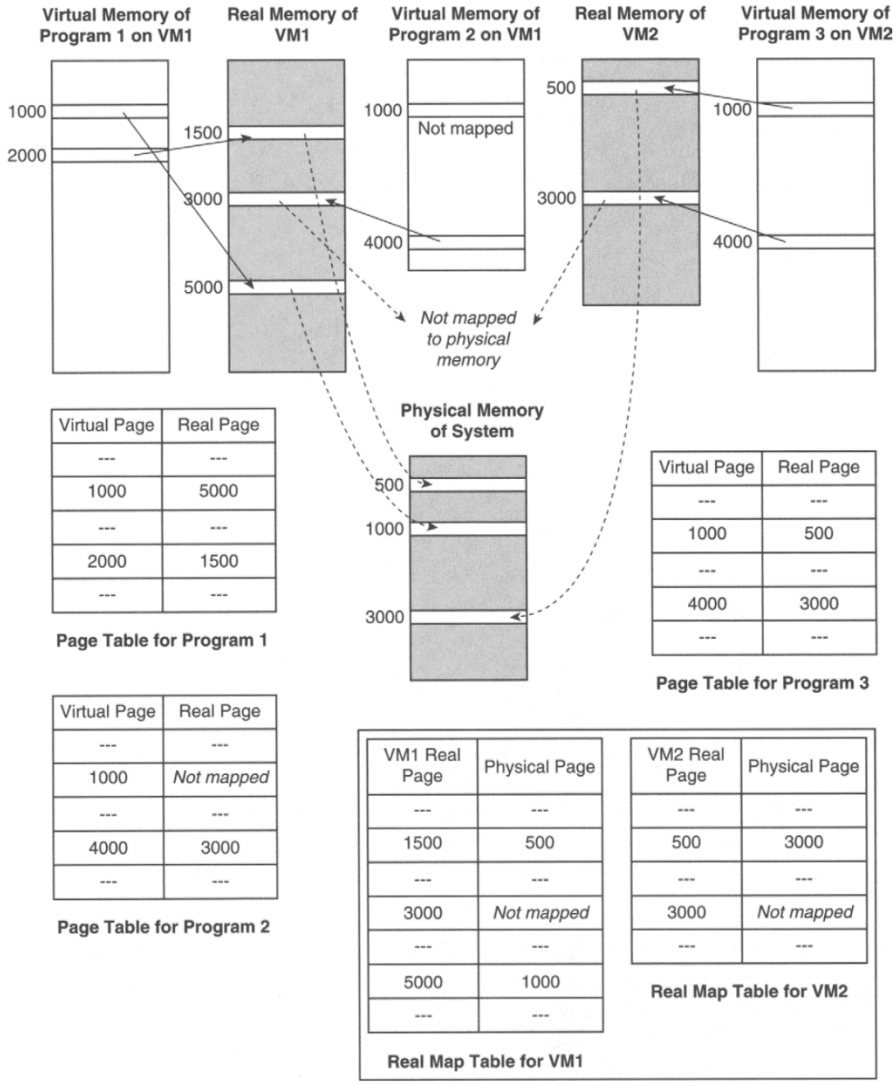
# 内存隔离

Xen将这层中间地址真正地映射到机器地址上却可以是不连续的，这样保证了所有的物理内存可被任意分配给不同的Guest OS



# 内存虚拟化


VM2
Process2
Process1
VM1
Process2
Process1


Virtual Memory of Program 1 on VM1
Real Memory of VM1
Virtual Memory of Program 2 on VM1
Real Memory of VM2
Virtual Memory of Program 3 on VM2
1000
2000
1500
3000
5000
1000
Not mapped
4000
500
3000
1000
4000
Not mapped to physical memory
Physical Memory of System
500
1000
3000
Virtual Page | Real Page
--- | ---
1000 | 5000
--- | ---
2000 | 1500
--- | ---
Page Table for Program 1
Virtual Page | Real Page
--- | ---
1000 | Not mapped
--- | ---
4000 | 3000
--- | ---
Page Table for Program 2
Virtual Page | Real Page
--- | ---
1000 | 500
--- | ---
4000 | 3000
--- | ---
Page Table for Program 3
VM1 Real Page | Physical Page
--- | ---
1500 | 500
--- | ---
3000 | Not mapped
--- | ---
5000 | 1000
--- | ---
Real Map Table for VM1
VM2 Real Page | Physical Page
--- | ---
500 | 3000
--- | ---
3000 | Not mapped
--- | ---
Real Map Table for VM2

# 内存保护机制

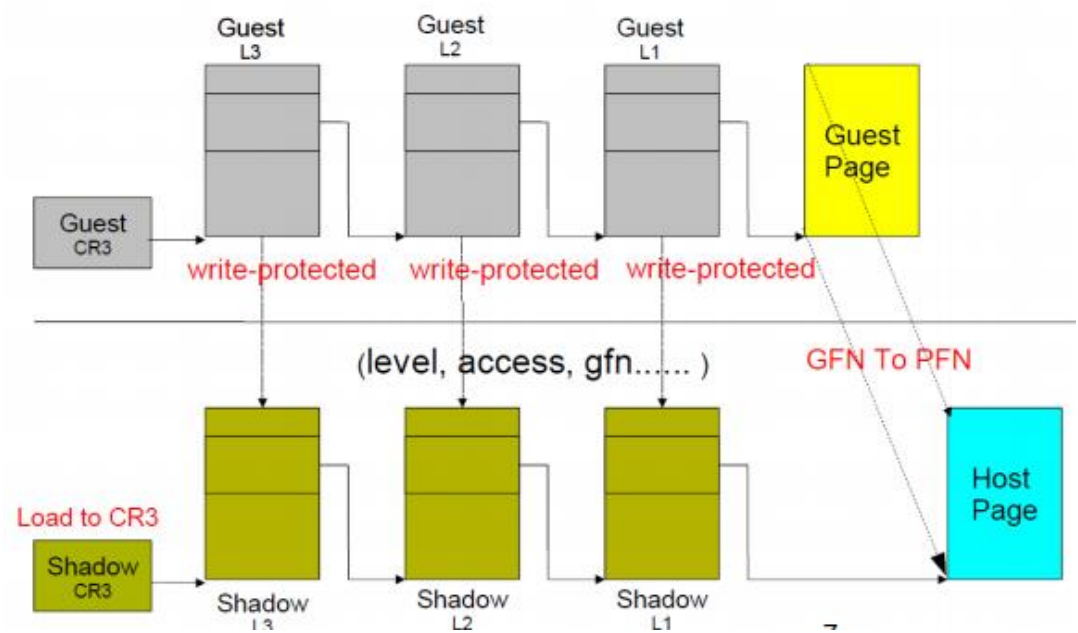
虚拟机监控器使用分段和分页机制对自身的物理内存进行保护。x86体系结构提供了支持分段机制的虚拟内存，这能够提供另一种形式的特权级分离。

| 基址 | 段限 | 属性位 |
| --- | --- | --- |
| 基址和虚拟地址相加形成线性地址 | 段限决定了这个段中所能访问的线性空间的长度 | 属性位则标记了该段是否可读写，可执行，是代码段还是数据段等 |

# 内存迁移

- OS在页表中配置GVA到GPA的映射，由于虚拟机内只能使用GPA，所以它将此页表的客户物理地址写入CR3
- VMM通过软件模拟方式使得CR3写入操作引发虚拟机下陷
- VMM为每个VM维护一个地址转换表，其中记录了GPA到HPA的映射关系。VMM遍历客户OS要安装的页表，并根据表中内容创建一个影子页表， 其包含了客户虚拟地址到主机物理地址的映射

  - VMM通过配置影子页表，将原页表所在的内存设置为只读
  - 影子页表的主机物理地址被VMM写入CR3.
  - VMM恢复虚拟机的执行
  - 在此之后，虚拟机使用的任何客户虚拟地址，将通过影子页表翻译为主机物理地址。



# 网络隔离

- 目标：确保把有害的攻击隔离，在可信网络之外和保证可信网络内部信息不外泄的前提下，完成网间数据的安全交换。

**关键**

**在于系统对通信数据的控制**

即通过不可路由的协议来完成网间的数据交换

**安全要素**

机密性 | 完整性 | 可用性 | 可控性 | 抗抵赖

**安全机制**

访问控制 | 身份认证 | 加密签名

# 网络隔离

**实现原理**

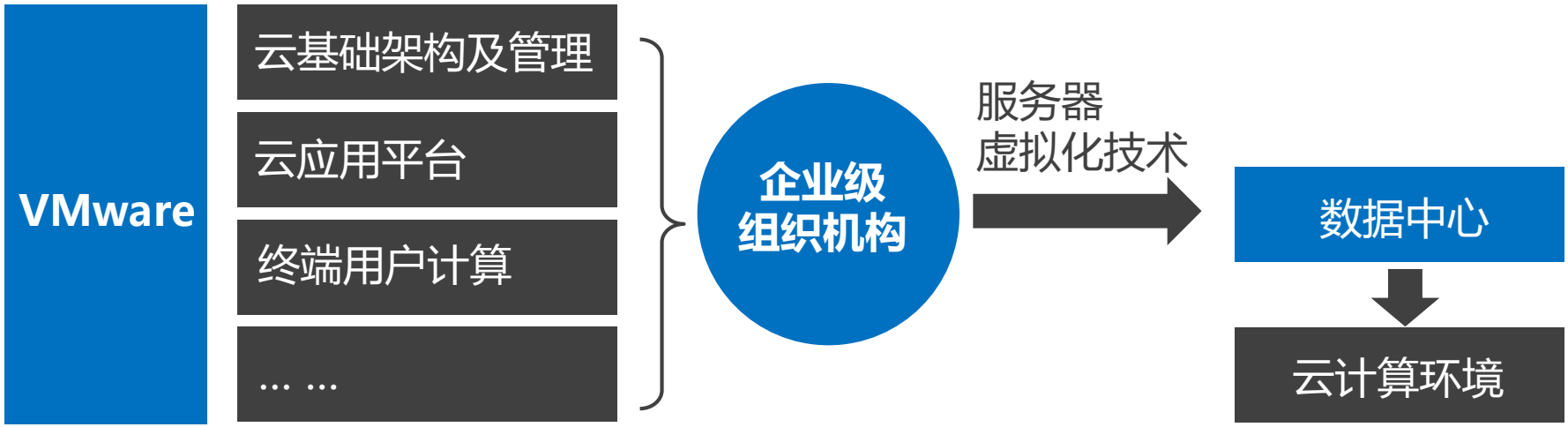
- 专用通信设备
- 专有安全协议
- 加密验证机制
- 应用层数据提取和鉴别认证技术

不同安全级别网络之间的数据交换

- 彻底阻断了网络间的直接TCP/IP连接
- 保证了网间数据交换的安全、可控
- 杜绝了由于操作系统和网络协议自身漏洞带来的安全风险

# Vmware虚拟化案例

VMware公司推出了面向云计算的一系列产品和解决方案




**虚拟机迁移工具VMotion**

**存储迁移工具Storage VMotion**

# VMotion

**VMotion**是VMware用于在数据中心的服务器之间进行虚拟机迁移的技术。

虚拟机迁移过程中主要采用**三项技术**：

① 将虚拟机状态信息压缩存储在共享存储器的文件中

② 将虚拟机的动态内存和执行状态通过高速网络在源ESX服务器和目标ESX服务器之间快速传输

③ 虚拟化网络以确保在迁移后虚拟机的网络身份和连接能保留

# VMware Storage VMotion

VMware Storage VMotion用于实时迁移虚拟机磁盘文件，以便满足对虚拟机磁盘文件的升级、维护和备份。

## 原理

**存储之间的转移**

## 核心技术

- **磁盘快照**
- **REDO记录**
- **父/子磁盘关系**
- **快照整合**

# 3

# 存储虚拟化

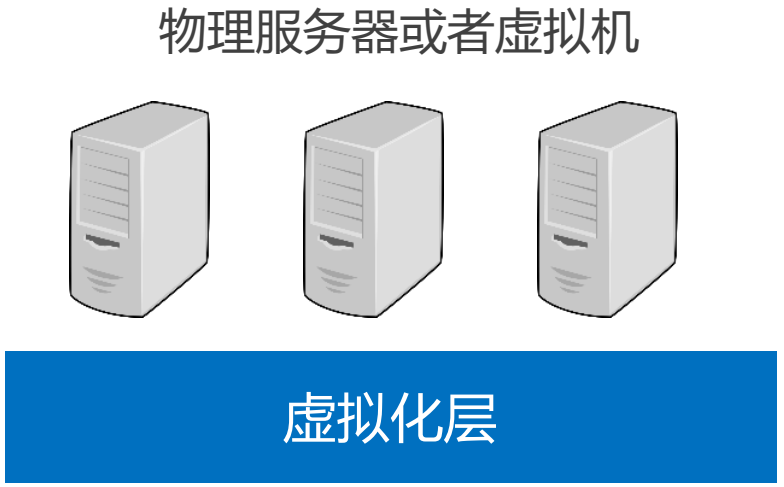
# 存储虚拟化关键技术

| |
|---|
| **存储虚拟化将系统中分散的存储资源整合起来** |
| 提高了存储资源利用率 |
| 降低了单位存储空间的成本 |
| 降低了存储管理的负担和复杂性 |
| **在虚拟层通过使用数据镜像、数据校验和多路径等技术** |
| 提高了数据的可靠性及系统的可用性 |
| **利用负载均衡、数据迁移、数据块重组等技术** |
| 提升系统的潜在性能 |
| **整合和重组底层物理资源** |
| 得到多种不同性能和可靠性的新的虚拟设备 |
| 满足多种存储应用的需求 |

# 存储虚拟化模型

一般来说，虚拟化存储系统在原有存储系统结构上增加了虚拟化层，将多个存储单元抽象成一个虚拟存储池，存储单元可以是异构，可以是直接的存储设备，也可以是基于网络的存储设备或系统。

- 减少存储系统的管理开销
- 实现存储系统数据共享
- 提供透明的高可靠性和可扩展性。



# 存储虚拟化的实现

**目前，实现存储虚拟化的方式主要有三种：**

基于**主机**的
存储虚拟化

基于**存储设备**
的存储虚拟化

基于**网络**的
存储虚拟化

# 基于主机的存储虚拟化

基于主机的存储虚拟化，也称基于服务器的存储虚拟化或者基于系统卷管理器的存储虚拟化，其一般是通过逻辑卷管理来实现的。

数据存储共享 | 存储资源管理 | 数据复制及迁移 | 集群系统 | 远程备份 | 灾难恢复

**优势**

- 性价比比较高

**劣势**

- 性能下降
- 可扩展性差
- 不支持异构平台

# 基于存储设备的存储虚拟化

基于存储设备的存储虚拟化主要是在存储设备的磁盘、适配器或者控制器上实现虚拟化功能。

- 有很多的存储设备的内部都有功能比较强的处理器
- 存储设备带有专门的嵌入式系统，可以在存储子系统的内部进行存储虚拟化，对外提供虚拟化磁盘
- 这类存储子系统与主机无关，对系统性能的影响比较小，也比较容易管理
- 对于包含有多家厂商提供异构的存储设备的SAN存储系统，基于存储设备的存储虚拟化方法的效果不是很好
- 这种设备往往规模有限并且不能进行级联，这就使得虚拟存储设备的可扩展性比较差

# 基于网络存储虚拟化

基于网络的存储虚拟化方法是在网络设备上实现存储虚拟化功能，包括基于互连设备和基于路由器两种方式。

**优点**

相对于上述几种方式，基于路由器的虚拟化在性能、效果和安全方面都要好一些

**缺点**

如果连接主机到存储网络的路由器出现故障，也可能会使主机上的数据不能被访问，但是只有与故障路由器连接在一起的主机才会受到影响，其余的主机还是可以用其他路由器访问存储系统，且路由器的冗余还能够支持动态多路径。

# 4

# 网络虚拟化

# 网络虚拟化

## 传统的数据中心

服务器之间操作系统和上层软件异构、接口与数据格式不统一

数据中心内网络传输效率低

## 使用云计算后

数据同步传送的大流量、备份大流量、虚拟机迁移大流量

采用统一的交换网络减少布线、维护工作量和扩容成本

## 引入虚拟化技术之后

数据中心网络虚拟化分为**核心层**、**接入层**和**虚拟机**网络虚拟化三个方面

# 核心网络虚拟化

核心层网络虚拟化，主要指的是数据中心核心网络设备的虚拟化。

**要求**

- 核心层网络具备超大规模的数据交换能力
- 足够的万兆接入能力

提供虚拟机箱技术

简化设备管理

提高资源利用率

提高交换系统的灵活性和扩展性

为资源的灵活调度和动态伸缩提供支撑

# 接入层网络虚拟化

接入层虚拟化，可以实现数据中心接入层的分级设计。根据数据中心的走线要求，接入层交换机要求能够支持各种灵活的部署方式和新的以太网技术。

**拥塞通知**
(IEEE 802.1Qau)

**增强传输选择ETS**
(IEEE 802.1Qaz)

**优先级流量控制PFC**
(IEEE 802.1Qbb)

**链路发现协议LLDP**
(IEEE 802.1AB)

# 虚拟机网络虚拟化

扩展虚拟数据中心中交换机和虚拟网卡的功能

802.1Qbg EVB （Edge Virtual Bridging）

802.1Qbh BPE（Bridge Port Extension）

**802.1Qbg**

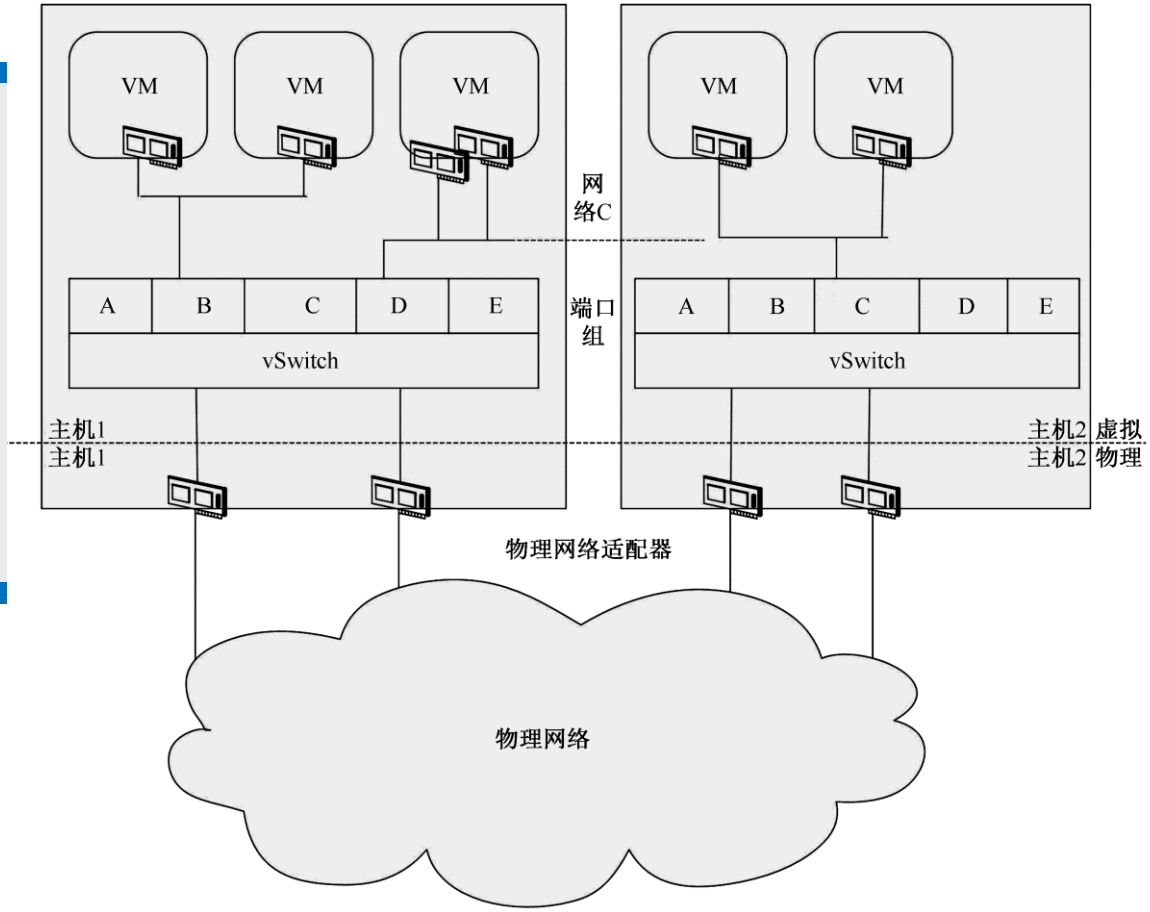
外部网络能够支持虚拟交换功能，对于虚拟交换网络范围内VM动态迁移、调度信息，均通过LLDP扩展协议得到同步以简化运维

**802.1Qbh**

将远程交换机部署为虚拟环境中的策略控制交换机，而不是部署成邻近服务器机架的交换机，通过多个虚拟通道，让边缘虚拟桥复制帧到一组远程端口

# VMware网络虚拟化

VMware的网络虚拟化技术主要是通过VMware vSphere中的vNetwork网络元素实现的，其虚拟网络架构如图所示。



# 虚拟网络接口

- 每个虚拟机都可以配置一个或者多个虚拟网络接口卡vNIC。
- 安装在虚拟机上的客户操作系统和应用程序利用通用的设备驱动程序与vNIC进行通信。
- 在虚拟机的外部，vNIC拥有独立的MAC地址以及一个或多个IP地址，且遵守标准的以太网协议。

# 虚拟交换机

- 虚拟交换机用来满足不同的**虚拟机**和**管理界面**进行互连。
- 每台服务器都有自己的虚拟交换机。
- 虚拟交换机的一端是与虚拟机相连的**端口组**，另一端是与虚拟机所在服务器上的物理以太网适配器相连的**上行链路**。
- 虚拟机通过与虚拟交换机上行链路相连的**物理以太网适配器**与外部环境连接。
- 虚拟交换机可将其上行链路连接到多个物理以太网适配器以**启用网卡绑定**。
- 通过网卡绑定，两个或多个物理适配器可用于**分摊流量负载**，或在出现物理适配器硬件故障或网络故障时提供被动故障切换。

# 5

# 桌面虚拟化

# 桌面虚拟化

- 每个桌面镜像就是一个带有应用程序的操作系统，终端用户通过一个虚拟显示协议来访问他们的桌面系统。这样做的目的就是使用户的使用体验同他们使用桌面上的PC一样
- 桌面虚拟化是一种基于中心服务器的计算机运作模型。


桌面虚拟化例子

# 桌面虚拟化技术现状

桌面虚拟化技术还面临着很多问题：

## 一 集中管理问题

虚拟化的服务器合并程度越高，此风险也越大。

## 二 集中存储问题

若是服务器出现了致命的故障，用户的数据可能丢失，整个平台将面临灾难。

## 三 虚拟化产品缺乏统一标准

各虚拟化产品厂商的产品间无法互通，一旦这个产品系列停止研发或其厂商倒闭，用户系统的持续运行、迁移和升级将会极其困难。

## 四 网络负载压力

如果用户使用的网络出现问题，桌面虚拟化发布的应用程序不能运行。

# VMWare桌面虚拟化

VMware View的主要部件如下：

## View Connection Server（View连接服务器）

接收到的远程桌面用户请求重定向到相应的虚拟桌面、物理桌面或终端服务器。

## View Manager Security Server（View安全连接服务器）

可选组件

## View Administrator Interface（View管理接口程序）

用于配置View Connection Server、部署和管理虚拟桌面、控制用户身份验证。

## View代理

安装在虚拟桌面依托的虚拟机、物理机或终端服务器上，安装后提供服务，可由View Manager Server管理。

# VMWare桌面虚拟化

## View Client（View客户端程序）

安装在需要使用“虚拟桌面”的计算机上，通过它可以与View Connection Server通信，从而允许用户连接到虚拟桌面。

## View Client with Offline Desktop（View 客户端程序）

支持View脱机桌面，可以让用户“下载”vSphere Server中的虚拟机到“本地”运行。

## View Composer

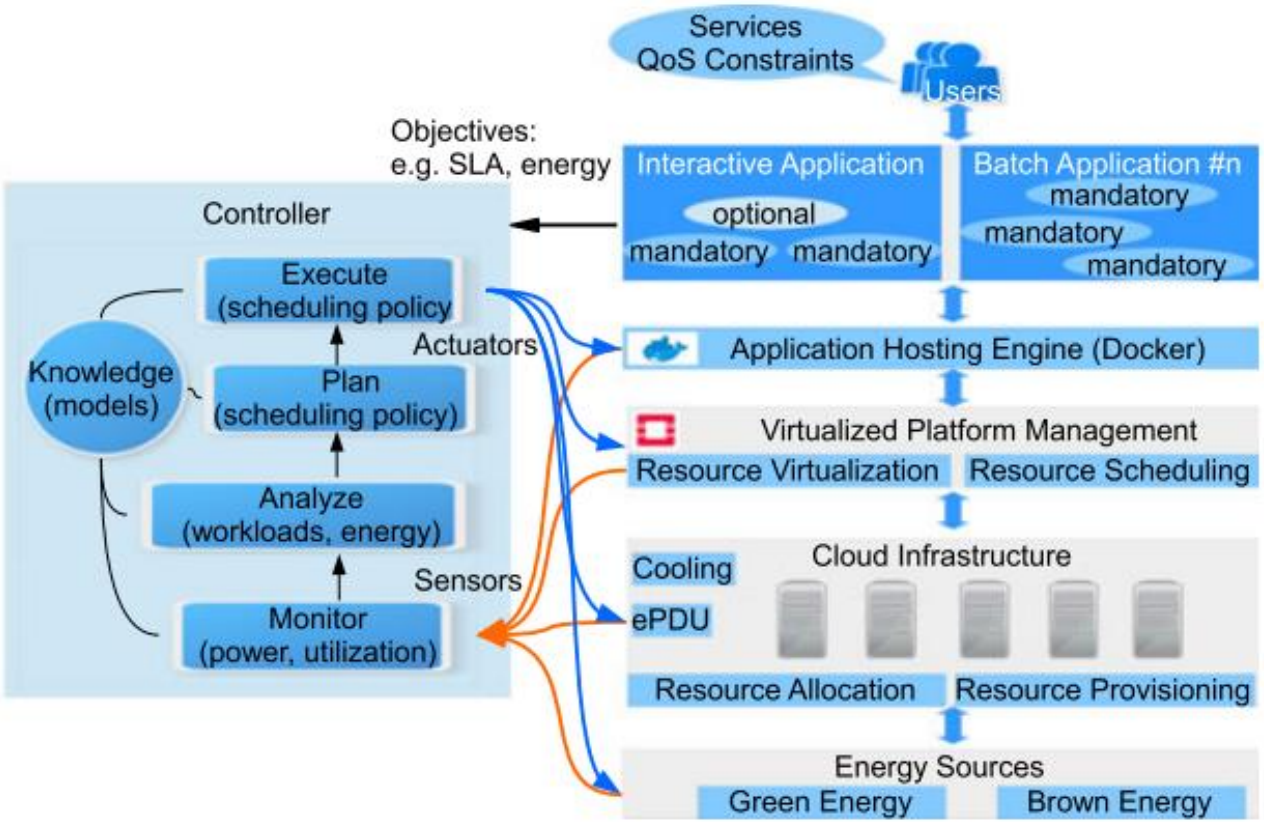
安装在vCenter Server上的软件服务，可以通过View Manager使用“克隆链接”的虚拟机

# 容器虚拟化

- 容器比虚拟机小得多或“轻”得多，通常由几兆字节组成，并且所需的硬件资源也少得多。这意味着一台物理服务器可以承载的容器比虚拟机要多得多。
- 容器可以在几秒甚至几毫秒内启动。相比之下，虚拟机的启动时间比较长。
- 由于容器都共享其主机的操作系统，因此所有应用程序都必须在同一操作系统上运行。相比之下，运行在虚拟主机上的虚拟机可以运行不同的操作系统（例如Linux，Unix和Windows）。
- 使用容器时，只需要对容器主机的操作系统进行补丁和更新。而虚拟机则需对每个操作系统都进行补丁和更新。
- 如果一个容器导致容器主机的操作系统崩溃，则在该主机上运行的所有容器都将失败。
- 容器主机的操作系统内核中的安全漏洞将影响其所托管的所有容器。

# 虚拟机管理例子

- 云数据中心多层级整体性节能调度框架




Ref: **M. Xu**, A. N. Toosi, R. Buyya, A Self-adaptive Approach for Managing Applications and Harnessing Renewable Energy for Sustainable Cloud Computing, ***IEEE Transactions on Sustainable Computing (T-SUSC)***, Vol. 6, Issue 4, pp. 544-558, Oct-Dec 2021.