

第五章 云使能技术

§5.1 宽带网络和Internet架构

§5.2 数据中心技术

§5.3 虚拟化技术

§5.4 Web技术

§5.5 多租户技术

§5.6 服务技术

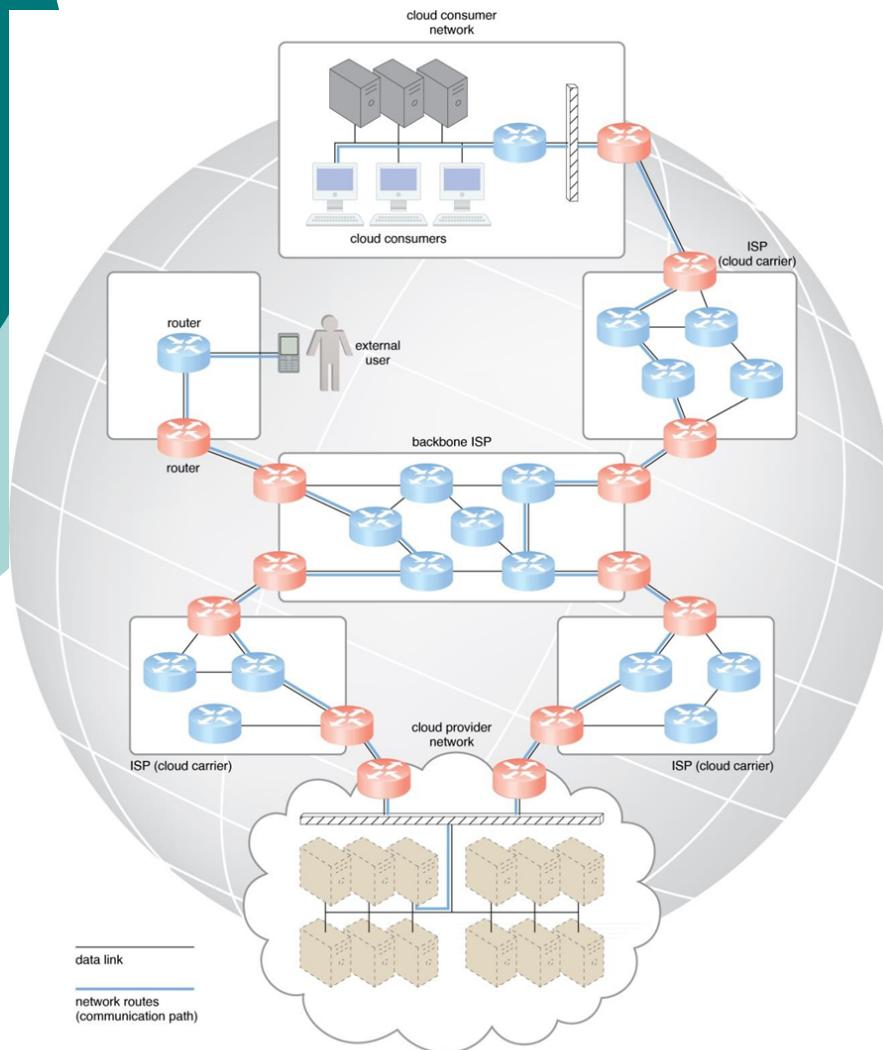


§5.1 宽带网络和Internet架构

- Internet服务提供者 (ISP)
 - 无连接分组交换(connectionless packet switching) (数据报网络)
 - 基于路由器的互联(router-based interconnectivity)
- 技术和商业考量
 - 连接性问题
 - 网络带宽和延迟问题
 - 云运营商和云提供者选择



Internet服务提供者 (ISP)



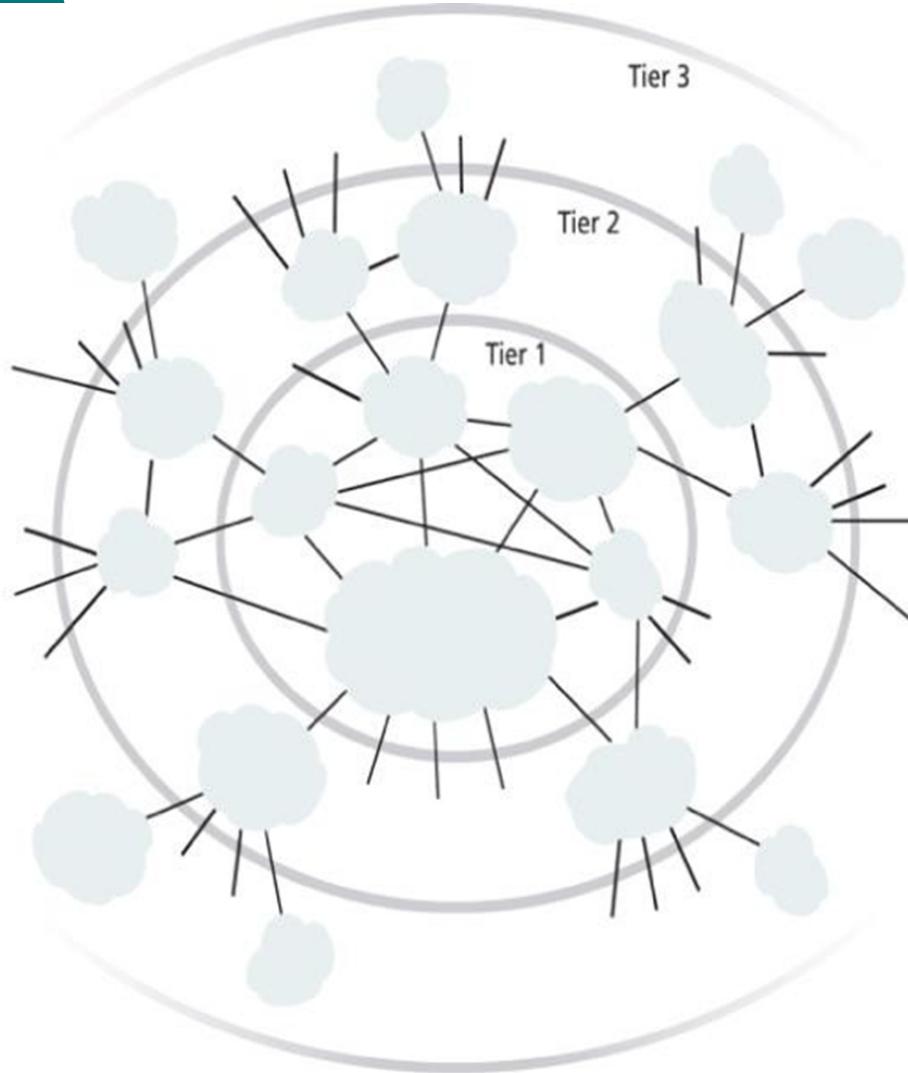
Messages travel over dynamic network routes in this ISP internetworking configuration.

The concept of the Internet was based on a decentralized provisioning and management model. ISPs can freely deploy, operate, and manage their networks in addition to selecting partner ISPs for interconnection. No centralized entity comprehensively governs the Internet.

一个ISP互联网络配置



Internet服务提供者 (ISP)



Worldwide connectivity is enabled through a **hierarchical topology** composed of Tiers 1, 2, and 3. The core Tier 1 is made of large-scale, international cloud providers that oversee massive interconnected global networks, which are connected to Tier 2's large regional providers. The interconnected ISPs of Tier 2 connect with Tier 1 providers, as well as the local ISPs of Tier 3. **Cloud consumers and cloud providers can connect directly using a Tier 1 provider, since any operational ISP can enable Internet connection.**

Internet互联结构的抽象示意图



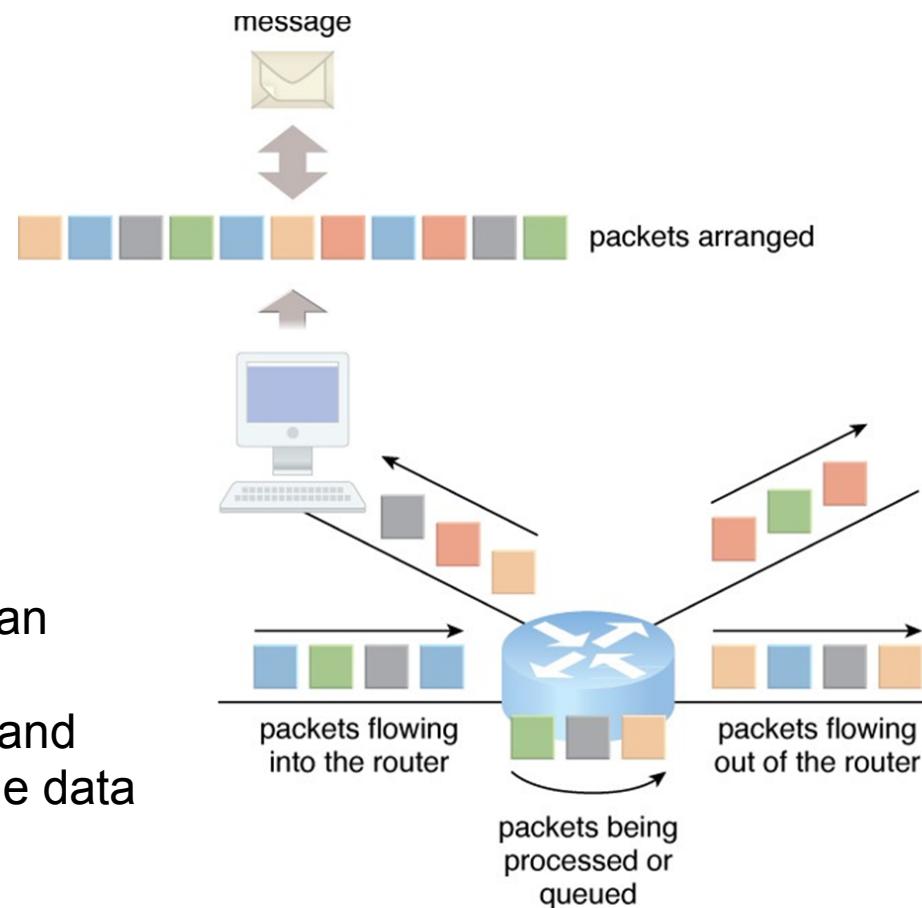
网络互联的基本机制

Definition

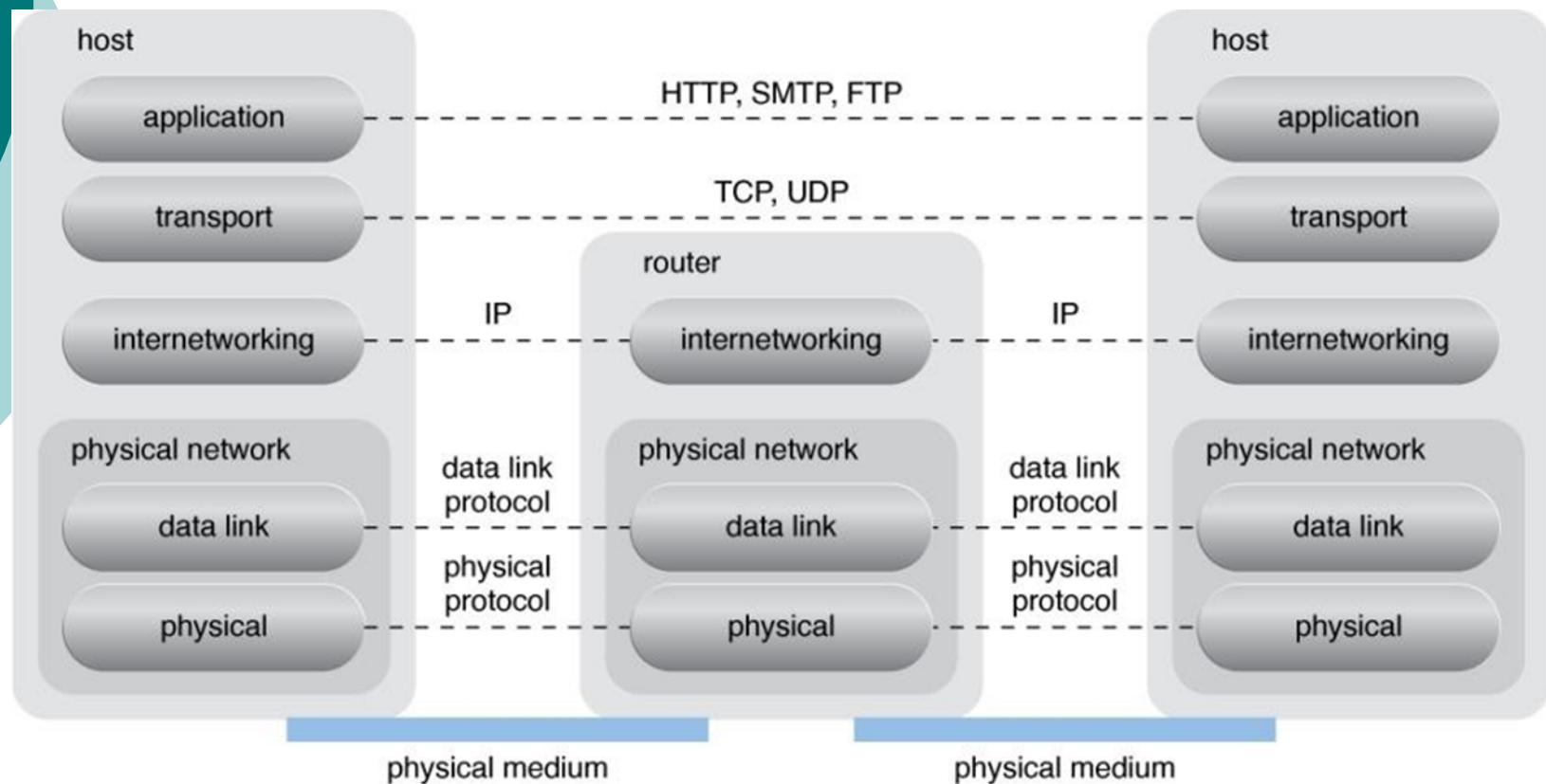
A router is a device that is connected to multiple networks through which it forwards packets.

Figure 5.3 – 路由器会指示数据包在Internet中的传递，还会把它们排列形成消息

A message is combined from an incoming group of disordered packets. The router receives and forwards packets from multiple data flows.



Internet参考模型和协议栈



Copyright © Arcitura Education

Figure 5.4 – 通用Internet模型和协议栈示意图



连接性问题---私有云的情况

In **traditional, on-premise** deployment models, enterprise applications and various IT solutions are commonly hosted on **centralized** servers and storage devices residing in the organization's **own data center**.

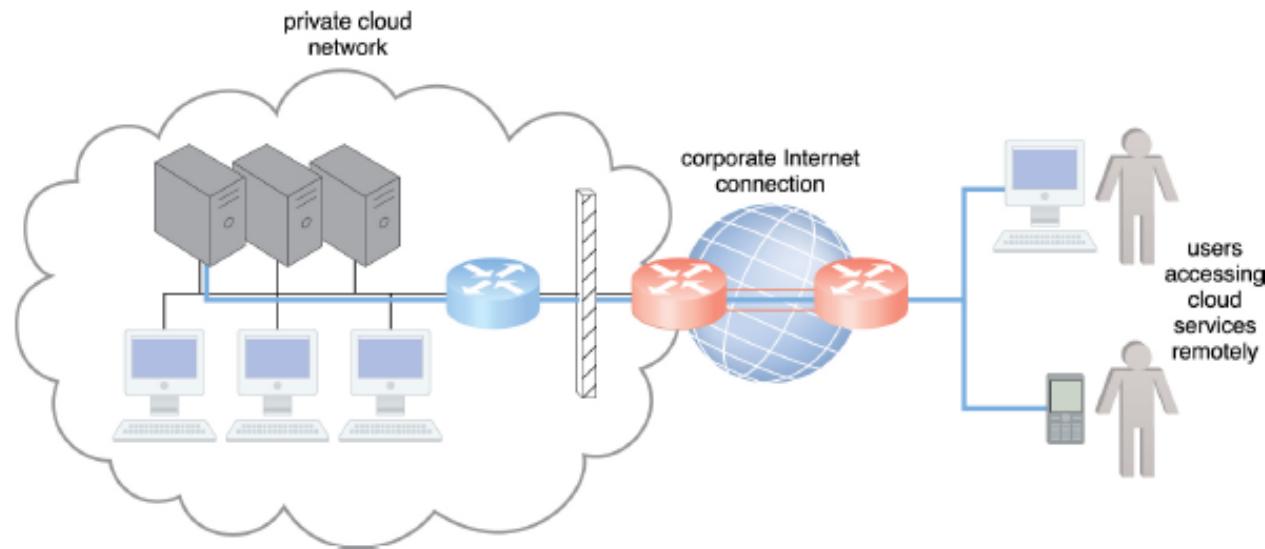


Figure : The internetworking architecture of a private cloud. The physical IT resources that constitute the cloud are located and managed **within the organization**.



连接性问题---基于Internet的情况

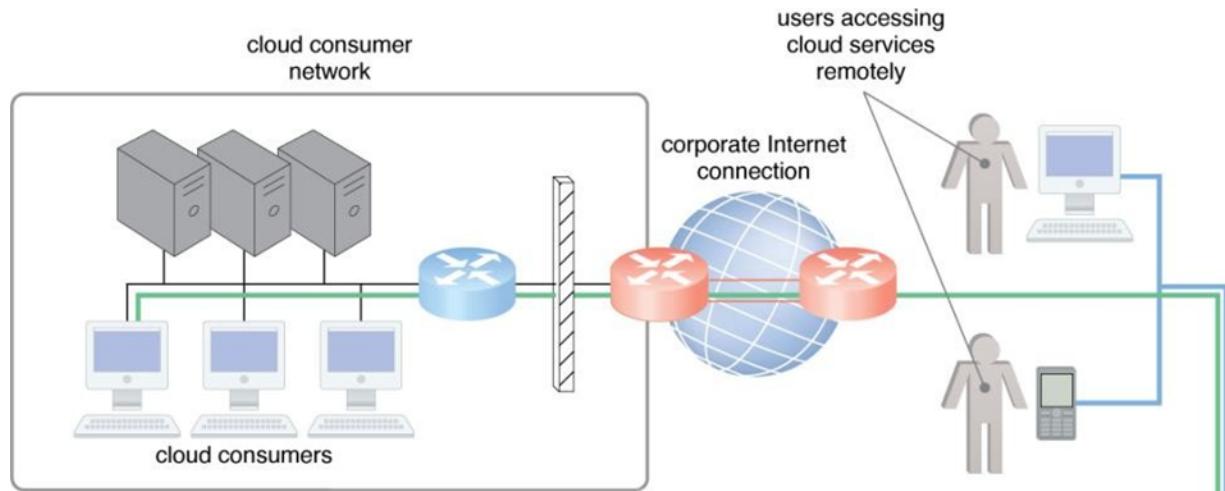
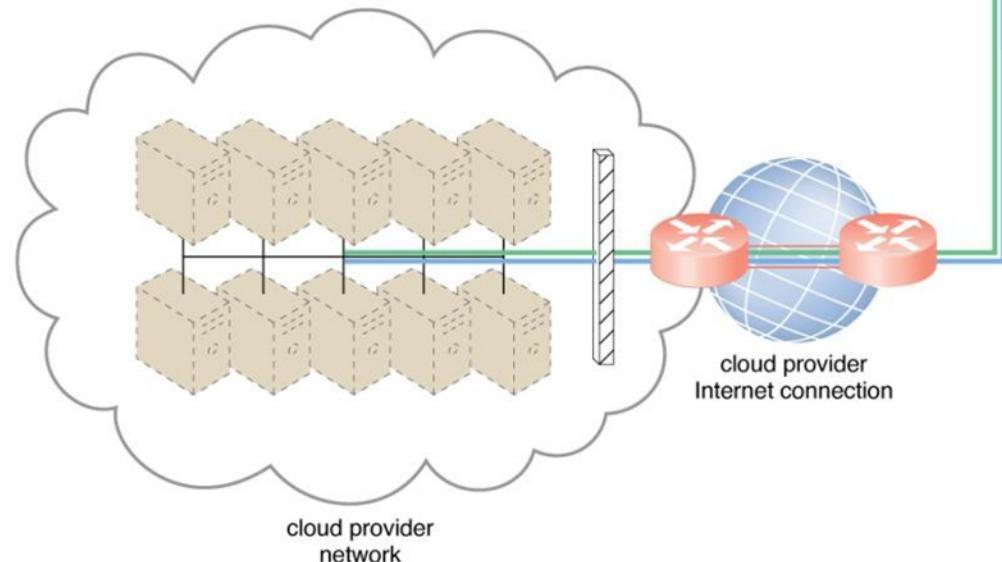


Figure 5.6 – 基于Internet的云部署模型的网络互联架构。

Internet是不直接相连的云用户、漫游终端用户和云提供者之间的网络连接代理



企业内部和基于云的网络互联的比较

企业内部IT资源	云IT资源
内部终端用户通过 企业网络 访问企业IT服务	内部终端用户通过 Internet连接 访问企业IT服务
内部用户在外网漫游时，通过 企业Internet连接 访问企业IT服务	内部用户在外网漫游时，通过 云提供者的Internet连接 访问企业IT服务
外部用户通过 企业Internet连接 访问企业IT服务	外部用户通过 云提供者的Internet连接 访问企业IT服务



关键点小结

- 云用户和云提供者通常利用**Internet**进行通信。**Internet**以去中心化的供给和管理模型为基础，不受任何集中式实体的控制。
- 网络互联架构的主要组件是使用网络路由器和交换机的无连接分组交换与基于路由器的互联。
- 网络带宽和延迟是影响**QoS**的因素，而网络拥塞对其有巨大影响。



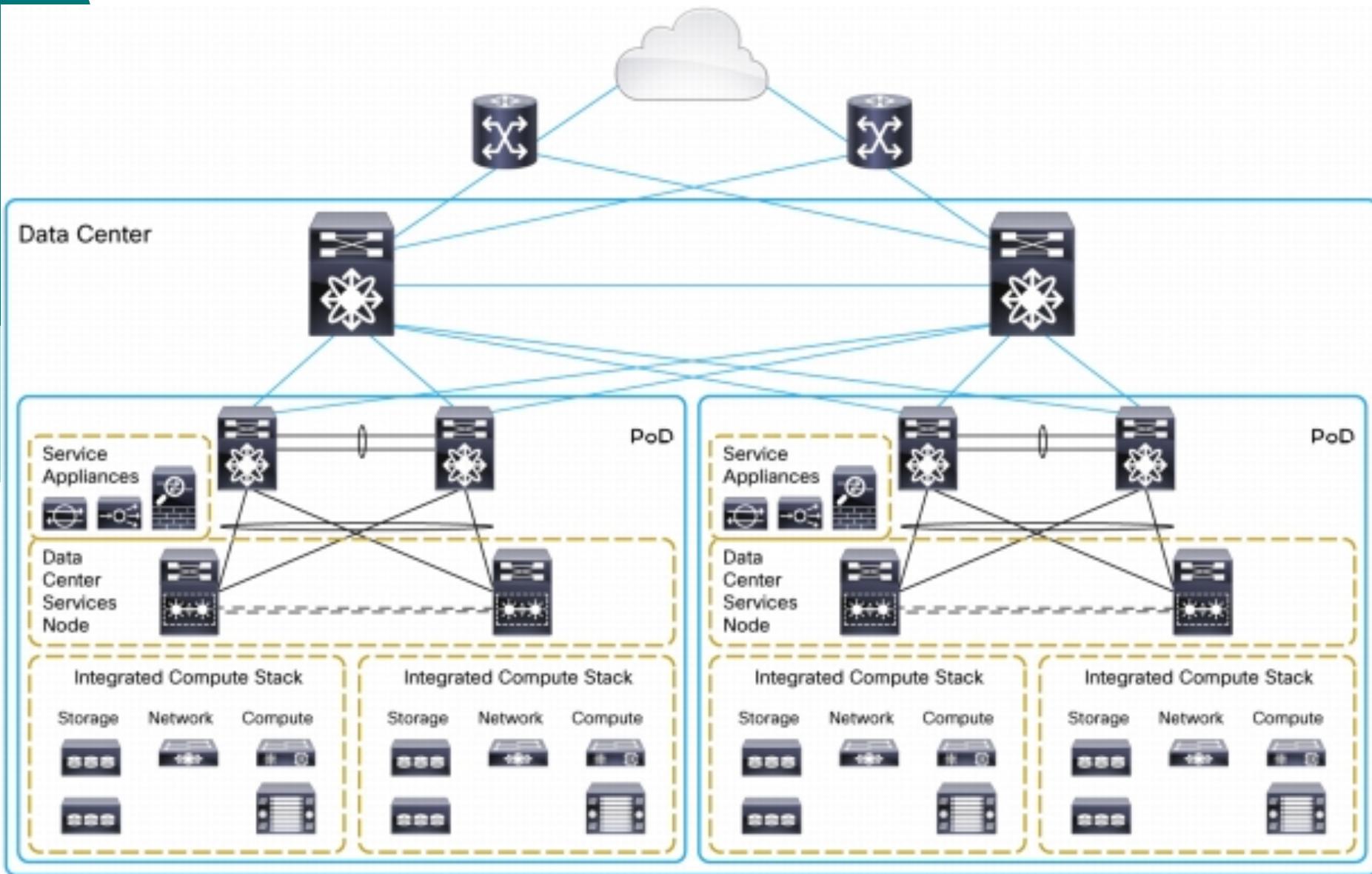
§5.2 数据中心技术

- 数据中心基本资源

- 云化的计算集群
- 硬件
 - 计算设备（标准化商用服务器）
 - 存储设备（磁盘阵列和存储虚拟化）
 - 网络设备（高速互联网络：以太网、光网等）
 - 配套设备（电源、散热……）
- 软件
 - 操作系统
 - 系统软件
 - 应用软件
 -



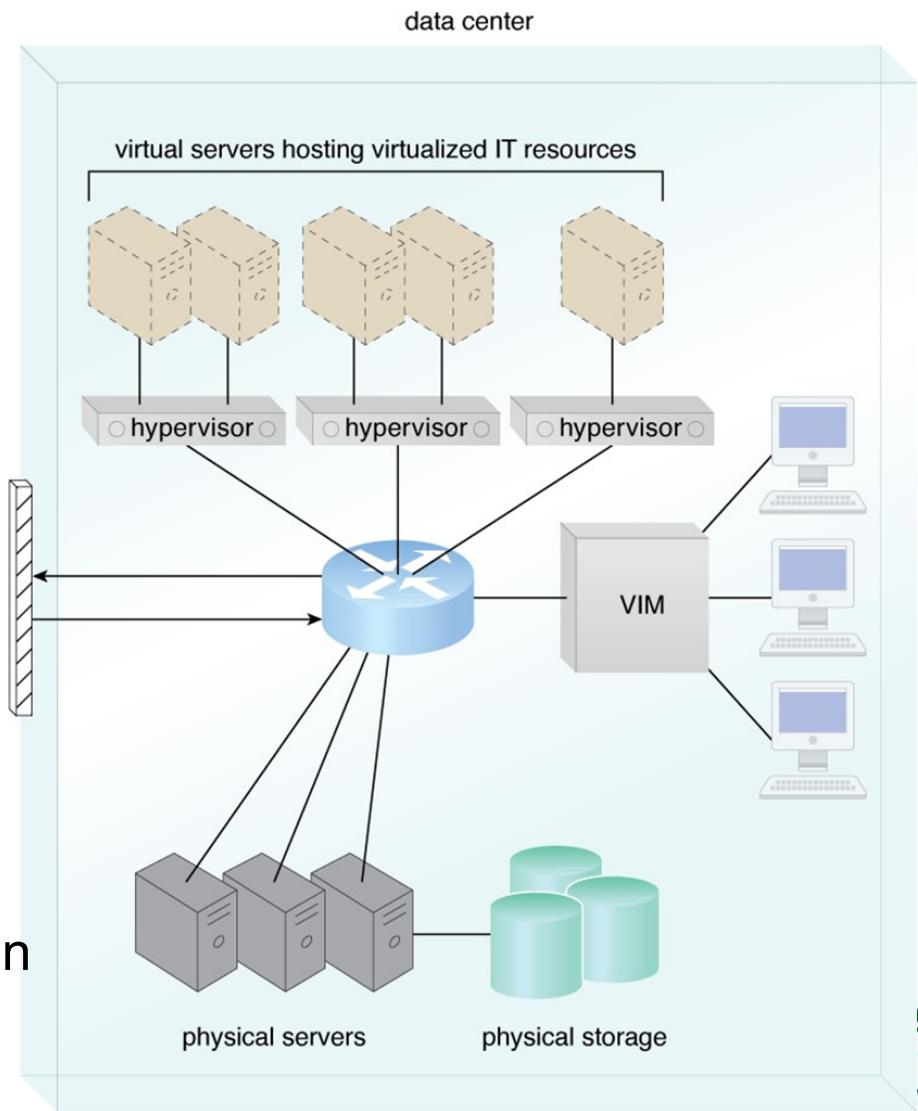
数据中心基本架构



数据中心基本技术

- 虚拟化
- 标准化与模块化
- 自动化（自配置和自恢复）
- 远程操作与管理
- 高可用性（采用高冗余度）

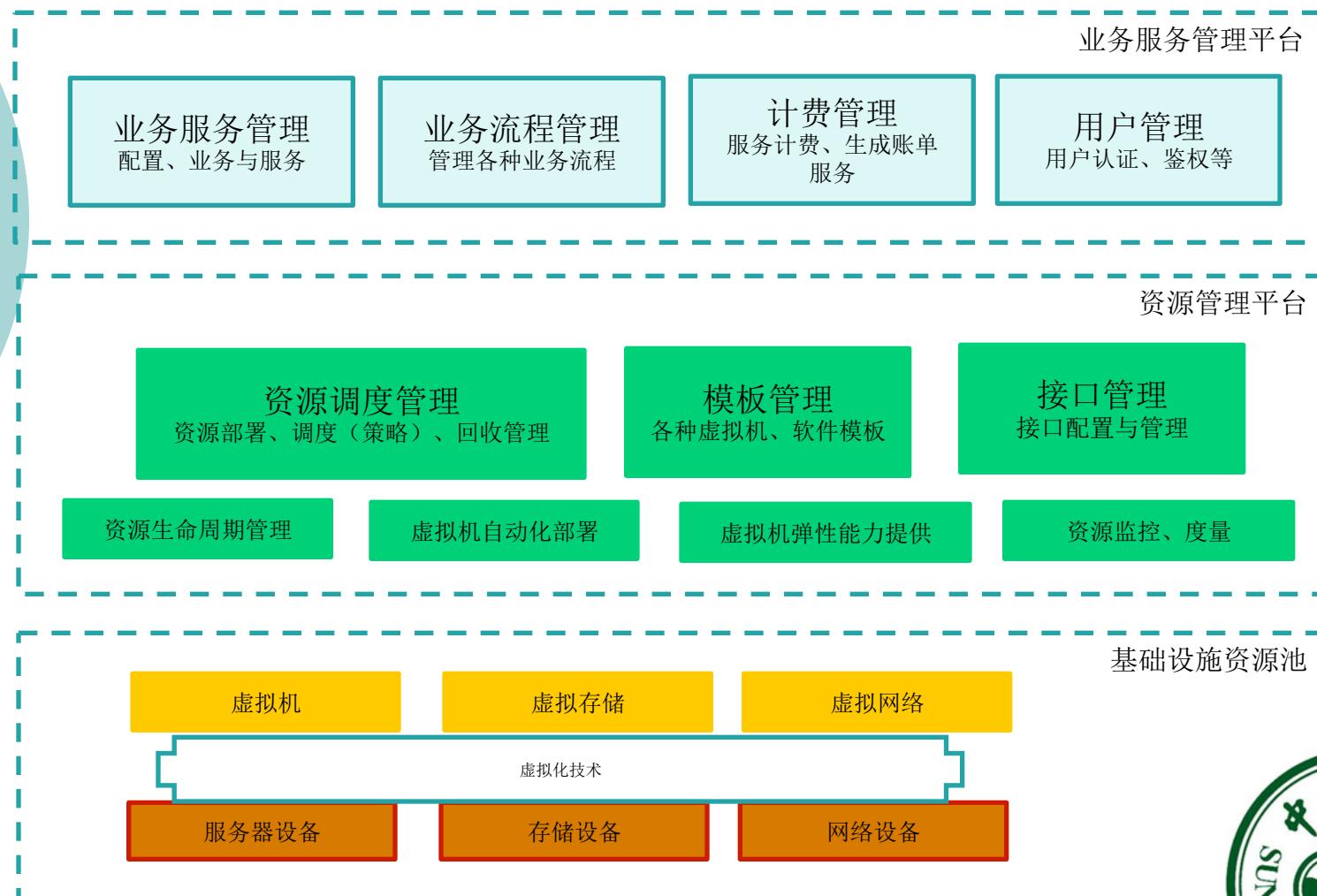
基础设施资源通过**VIM** (*virtualization infrastructure management*) 虚拟化成虚拟资源，并通过**hypervisor** 管理程序生成不同的虚拟服务器。



Copyright © Arcitura Education



数据中心管理软件层次（三层）



自动化部署

○ 初次部署

- 选择物理服务器：
 - 尽量不启动新服务器
 - CPU和I/O互补
- 挂接系统模板
 - 配置主机名、IP

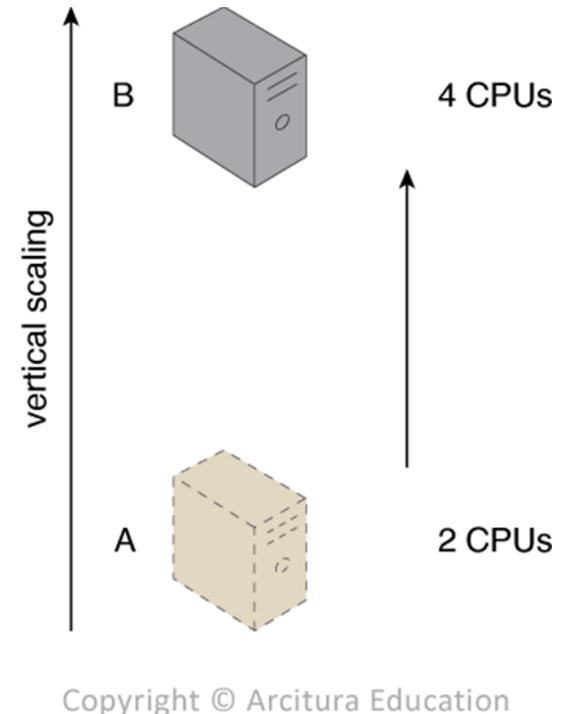
○ 迁移

- 虚拟机操作系统复制
- 动态迁移：边运行边迁移

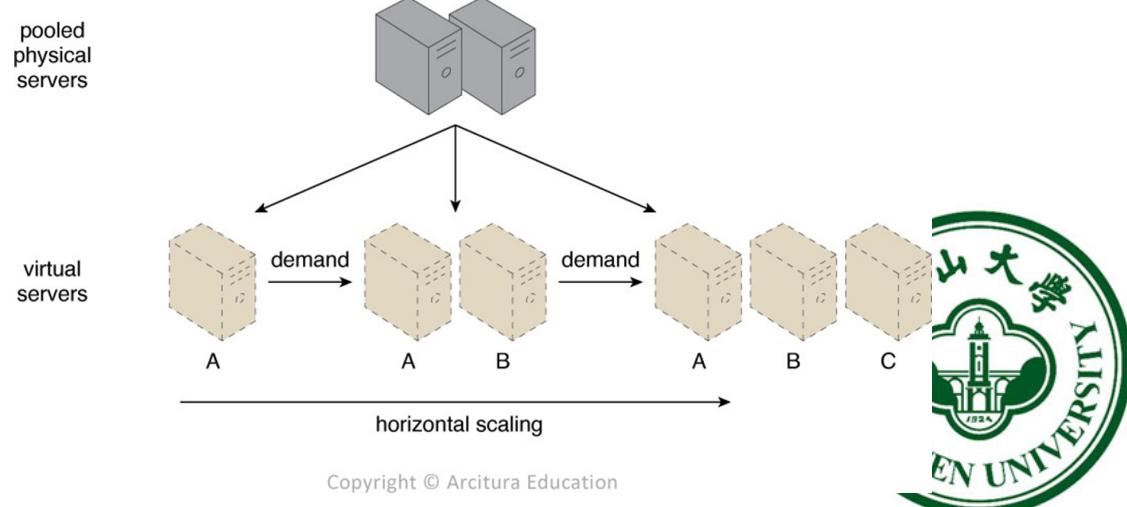


弹性能力提供

- 向上/下扩展 (Scale Up/Down)
 - 根据负载高低调整资源分配
 - 多用于小型系统
- 向内/外扩展 (Scale In/Out)
 - 根据负载升降增加或减少虚拟机数量
 - 负载均衡是关键
 - 多用于大型系统



Copyright © Arcitura Education



Copyright © Arcitura Education



资源监控

功能

- 状态监控
- 性能监控
- 容量监控
- 安全监控
- 使用度量

常用方法

- 日志分析
- 报警探
- 探针采集



资源调度

- 需求分析和预测
- 网络状况
- 负载均衡
 - 负载类型的互补
- 能耗

An ACO-Based Scheduling Strategy on Load Balancing in Cloud Computing Environment

Wei-Tao Wen*, Chang-Dong Wang[§], De-Shen Wu[†] and Ying-Yan Xie[‡]

School of Mobile Information Engineering, Sun Yat-sen University, Zhuhai, P. R. China, 519082
Email: *332974874@qq.com, [§]changdongwang@hotmail.com, [†]1515076287@qq.com, [‡]1807637279@qq.com

Abstract—Overload balance of cloud data centers is a matter of great concern. Live migration of virtual machines presents an effective method to realize load balancing and to optimize resources utilization. With the rapidly increasing scale of cloud data centers, traditional centralized migration strategy begins to show lack of scalability and reliability. In this paper, we propose a novel distributed VM migration strategy based on a metaheuristic algorithm called Ant Colony Optimization. In our ACO-VMM Strategy, local migration agent autonomously monitors the resource utilization and launches the migration. At monitoring stage, it takes both the previous and current system condition into account to avoid unnecessary migrations. Besides, it adopts two different traversing strategies for ants in order to find the near-optimal mapping relationship between virtual machines (VMs) and physical machines (PMs). Experimental results show that ACO-VMM outperforms the existing migration strategies by achieving load balance of whole system, as well as reducing the number of migrations and maintaining the required performance levels.

Index Terms—overload balance; distribution; migration strategy; ant colony optimization

be guaranteed [5]. Therefore, we need to find valid ways to schedule VM resources to realize load balancing in cloud computing and to optimize resource utility.

Over the years, there are many researches on VM scheduling strategy. One well-known strategy is centralized VM live migration scheduling scheme, as shown in Fig. 1. The scheduler mainly consists of two components: central controller and local migration controller. The central controller obtains all the physical resources utilization on the whole, and then initializes the VM migration based on pre-specified policies so as to achieve load balancing and high resource utilization. The local migration controller sends its physical resources utilization to center controller and receives instructions to start migration. However, researchers have found some defects of such framework like lack of reliability and scalability [6]. If the central controller breaks down, it will easily lead to load imbalance of the system. Moreover, the efficiency is another bottleneck of the entire system.



§5.3 虚拟化技术

虚拟化是指将物理IT资源转换为虚拟IT资源的过程。

大多数IT资源都能被虚拟化，包括：

- 服务器（server）——一个物理服务器可以抽象为一个虚拟服务器。
- 存储设备（storage）——一个物理存储设备可以抽象为一个虚拟存储设备或一个虚拟磁盘。
- 网络（network）——物理路由器和交换机可以抽象为逻辑网络，如VLAN。
- 电源（power）——一个物理UPS和电源分配单元可以抽象为通常意义上的虚拟UPS。

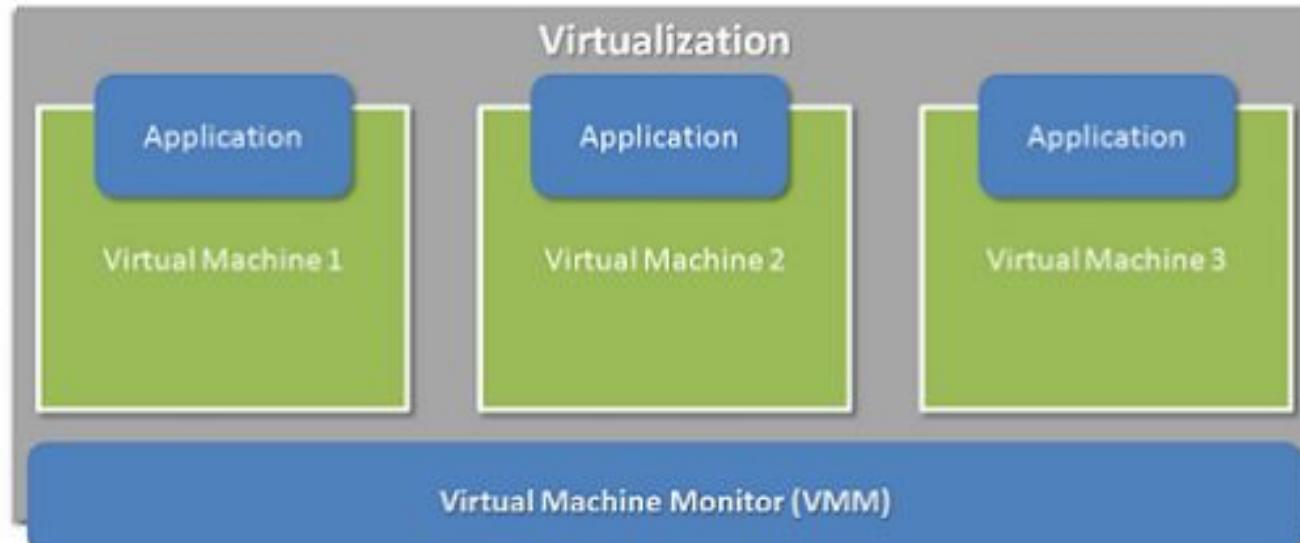


服务器虚拟化

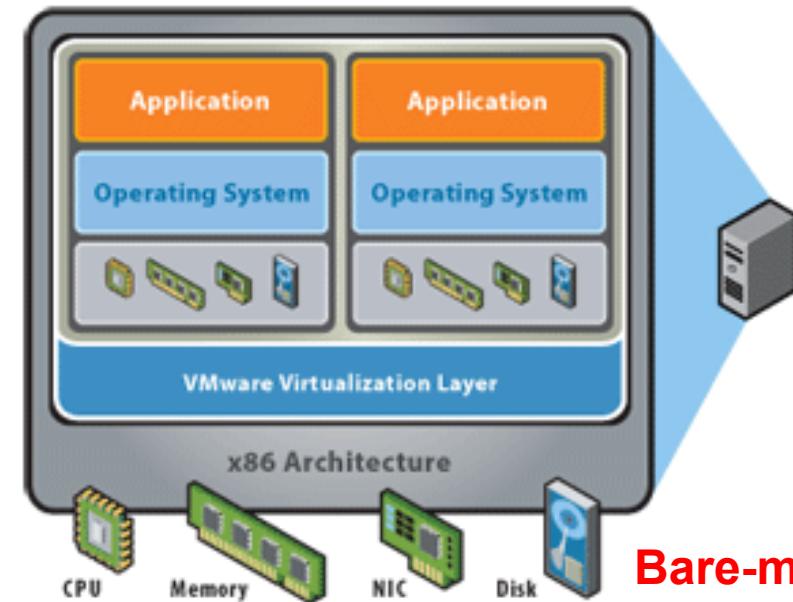
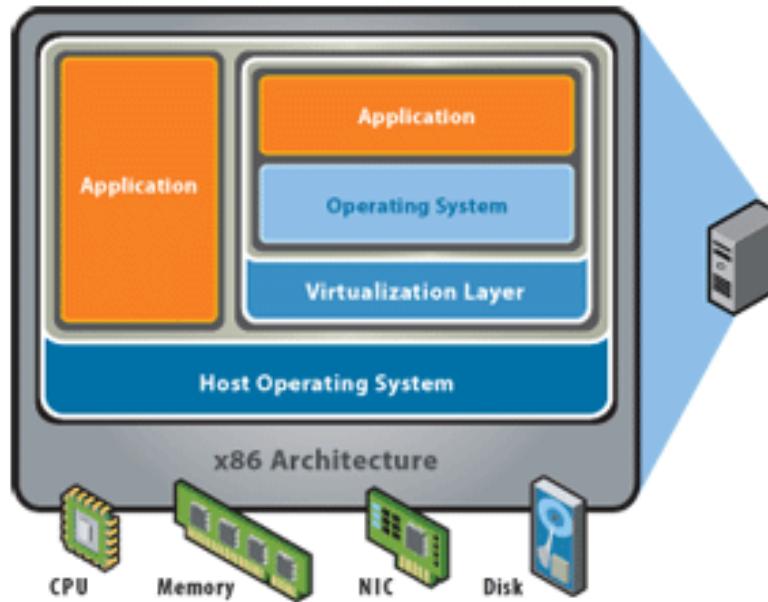
- 一台物理机 → 多台虚拟机 (Virtual Machine, VM)
 - 虚拟CPU
 - 虚拟内存
 - 虚拟I/O
 - 独立软件
 - 数据隔离



虚拟机架构-两种



基本架构



Hosted

Bare-metal

基于操作系统的虚拟化--寄生架构（Hosted）

Definition

Operating system-based virtualization is the installation of virtualization software in a **pre-existing operating system**, which is called the host operating system.

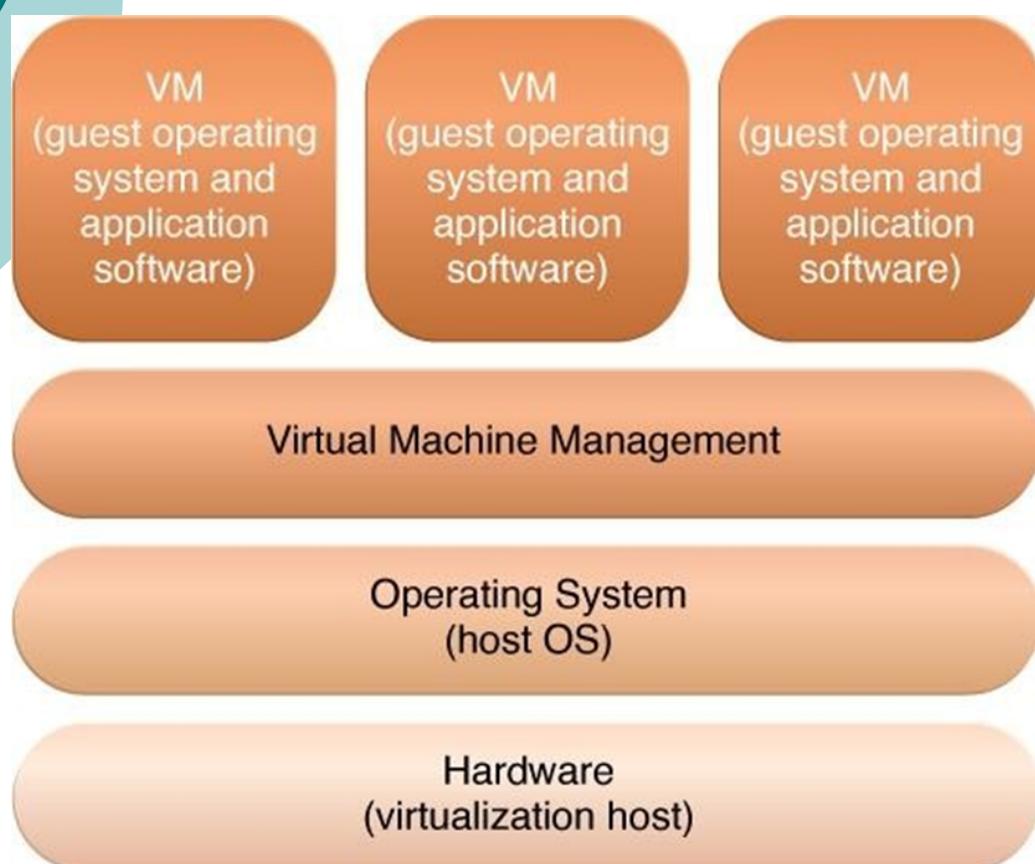


Figure 5.8 – 基于操作
系统虚拟化的逻辑
分层。其中，VMM
首先被安装在完整的
宿主操作系统上，然后
被用于产生虚拟机



基于硬件的虚拟化--裸金属架构 (Bare-metal)

Definition

Hardware-based virtualization represents the installation of virtualization software directly on the **physical host hardware** so as to bypass the host operating system, which is presumably engaged with operating system-based virtualization.

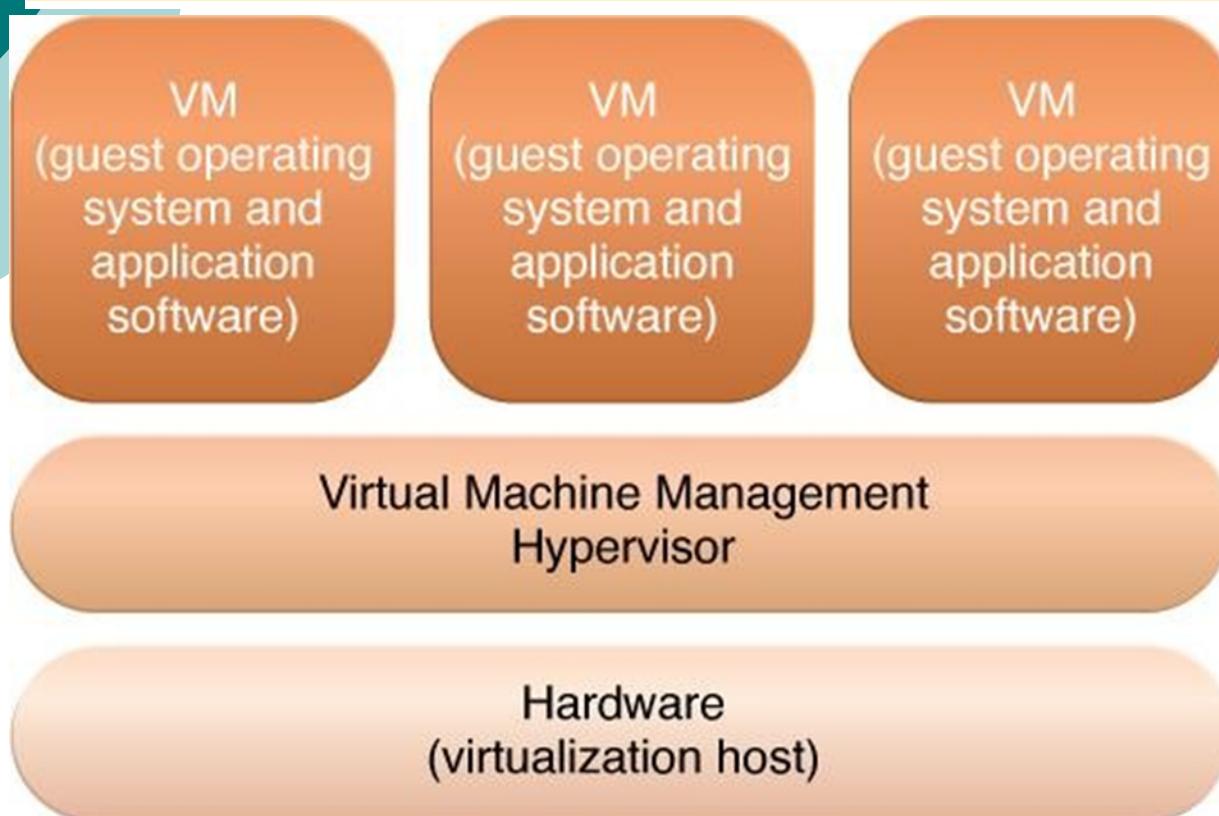
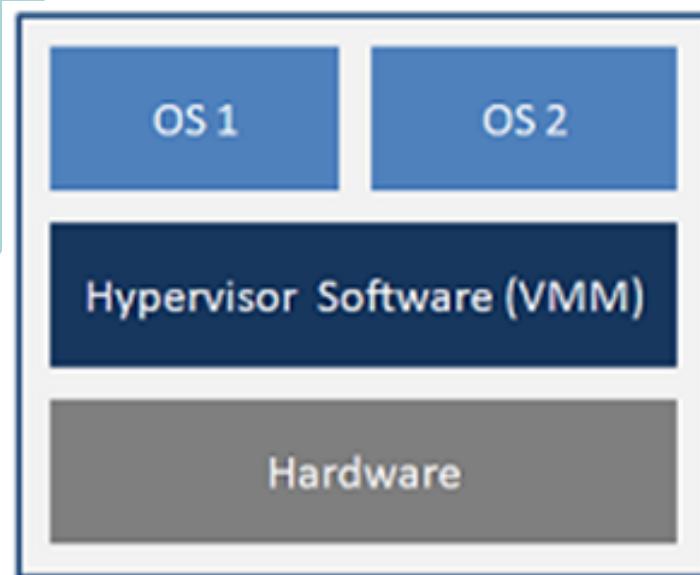


Figure 5.9 – 基于硬件虚拟化的逻辑分层，不再需要另一个宿主操作系统，而是将VMM直接安装在硬件上。

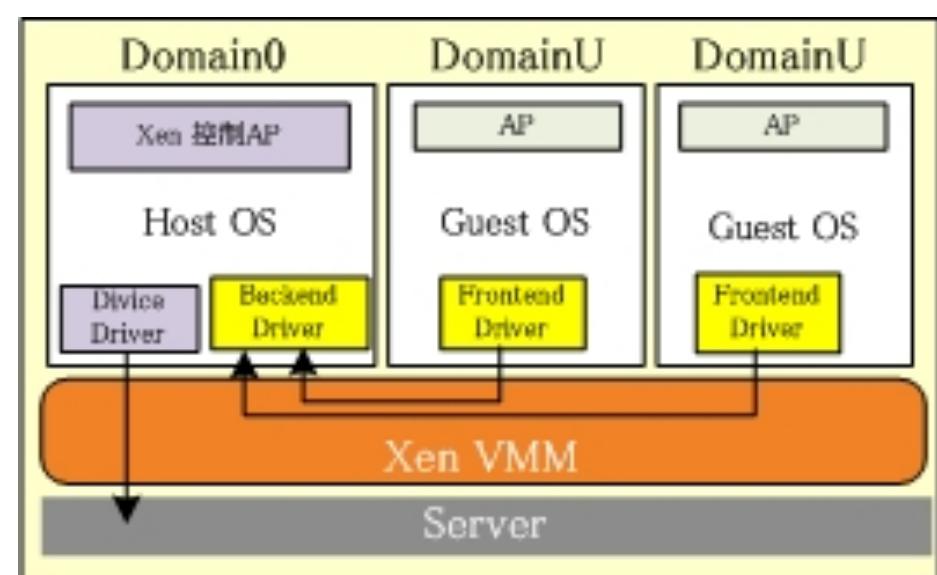


虚拟机架构 – 裸金属架构 (Bare-metal)

- VMM + OS



Bare-Metal Architecture



Hosted vs. Bare-metal

Bare-metal (Type I)	Hosted (Type II)
效率高 (不受Host OS影响、I/O优化)	
安全性高	
	使用方便
	功能丰富 (e.g. 3D加速)
适于服务器系统	适于桌面系统



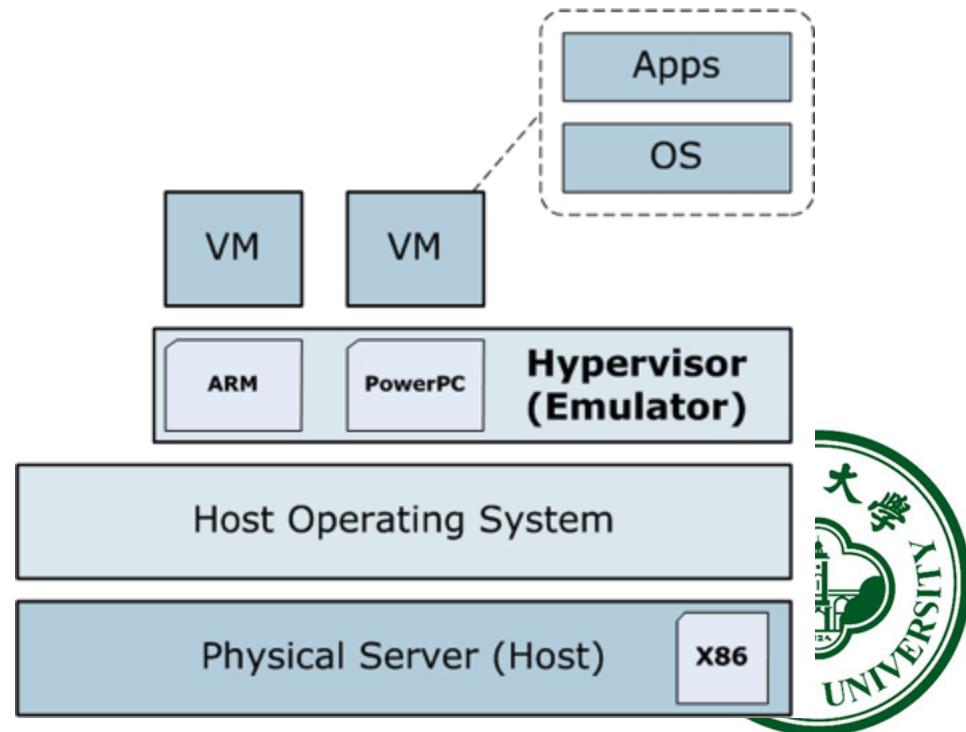
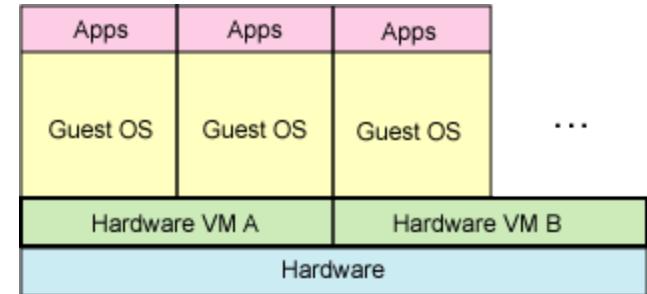
系统虚拟化技术

- 硬件仿真（**Emulation**）
- 全虚拟化（**Full Virtualization**）
- 半虚拟化（**Paravirtualization**）
- 硬件辅助虚拟化（**Hardware Assisted Virtualization**）
- 操作系统级虚拟化（**Operating System Level Virtualization**）



硬件仿真 (Emulation)

- 属于 Hosted 架构
- Host OS → VMM (Emulator) → Guest OS。
- 技术要点
 - 特权指令(Guest)>>陷入>>VMM>>模拟>>特权指令(Host)
- 知名产品
 - Bochs (open source)
 - QEMU (free)
 - Virtual PC (Microsoft)



硬件仿真 (Emulation)



Windows Virtual PC 是最新的 Microsoft 虚拟化技术。您可以使用此技术在一台计算机上同时运行多个操作系统，并且只需单击一下，便可直接在运行 Windows 7 的计算机上的虚拟 Windows 环境中运行许多生产应用程序。

Windows Virtual PC 支持以下主机操作系统和来宾操作系统：

- **主机**：Windows 7 家庭普通版、Windows 7 家庭高级版、Windows 7 专业版、Windows 7 旗舰版、Windows 7 企业版。
- **来宾**：Windows XP Service Pack 3 (SP3) Professional、Windows Vista Enterprise Service Pack 1 (SP1)、Windows Vista Ultimate Service Pack 1 (SP1)、Windows Vista Business Service Pack 1 (SP1)、Windows 7 专业版、Windows 7 旗舰版、Windows 7 企业版。

注意：在 Windows Vista Business Service Pack 1 (SP1) 和 Windows 7 专业版上不支持虚拟应用程序功能。

① 系统要求

支持的操作系统

Windows 7

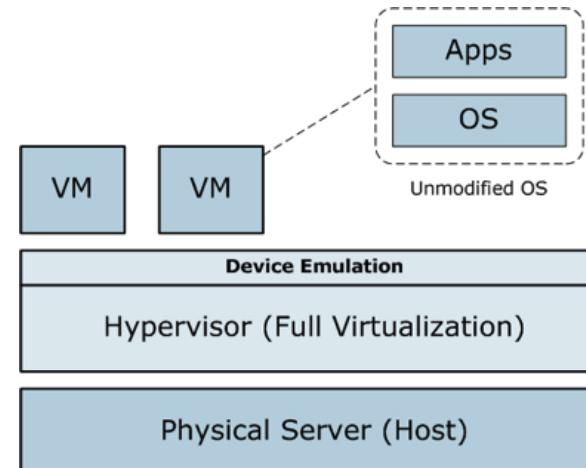
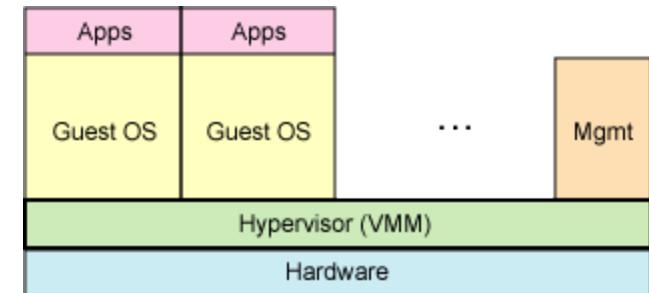
- **处理器**：支持硬件辅助虚拟化功能的 Intel®、AMD®、VIA® 处理器，并在 BIOS 中启用此设置。
- **内存**：2GB 内存（推荐）。
- **硬盘要求**：安装 Windows Virtual PC 需要 20MB 硬盘空间。

全虚拟化 (Full Virtualization)

Definition

Full virtualization is a virtualization technique used to provide a **complete simulation** of the underlying hardware.

- 最常见、最成熟的虚拟化技术
- Hosted 和 Bare-metal都有
- 技术要点：
 - 在客户操作系统和硬件之间捕捉和处理那些对虚拟化敏感的特权指令，使客户操作系统无需修改就能运行
- 知名产品
 - IBM CP/CMS
 - Oracle VirtualBox
 - KVM
 - VMware ESX



全虚拟化 (Full Virtualization)

安全 | <https://www.vmware.com/products/esxi-and-esx.html>

VMware

VMWARE CLOUD

PRODUCTS

SOLUTIONS

SUPPORT

DOWNLOADS

PROFESSIONAL SERVICES

PARTNER PROGRAMS

COMPANY

Products > ESXi

US Login Training Community Store 1-877-486-9273 Search

Twitter Facebook LinkedIn YouTube Google+

Contact Sales »

OVERVIEW FEATURES MANAGEMENT SCRIPTING FAQS GETTING STARTED

ESXi The Purpose Built Bare Metal Hypervisor

VMware ESXi is the industry-leading, purpose-built bare-metal hypervisor. ESXi installs directly onto your physical server enabling it to be partitioned into multiple logical servers referred to as virtual machines.

Customers can use ESXi with either the [free vSphere Hypervisor](#) or as part of a [paid vSphere edition](#).

vSphere ESXi Hypervisor Features

Improved Reliability and Security

The ESXi bare-metal hypervisor's management functionality is built into the VMkernel, reducing the footprint to 150 MB. This gives it a very small attack surface for malware and over-the-network threats, improving reliability and security.

Reduced Management Overhead

vSphere ESXi uses an agentless approach to hardware monitoring and system management with an API-based partner integration model. Management tasks are run on remote command lines with the vSphere Command Line Interface (vCLI) and Power CLI, which uses Windows PowerShell cmdlets and scripts for automated management.

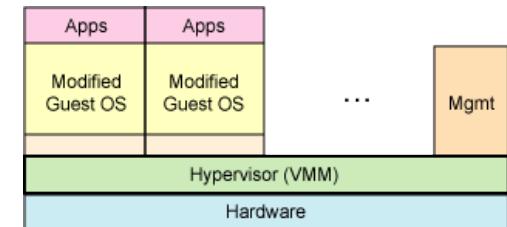
Streamlined Deployment and Configuration

vSphere Free Trial

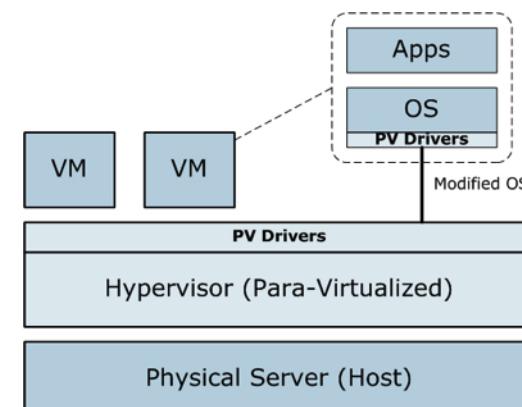
半虚拟化 (Para-virtualization)

Definition

In computing, para-virtualization is a virtualization technique that presents a software interface to virtual machines that is **similar, but not identical** to that of the underlying hardware.



- Bare-metal模式
- 技术要点：
 - 与全虚拟化类似,利用VMM实现对底层硬件的访问
 - Guest OS集成与半虚拟化有关的代码, 以配合VMM
 - 无需重新编译或捕获特权指令, 性能非常接近物理机
- 知名产品
 - Xen
 - Microsoft Hyper-V



半虚拟化 (Para-virtualization)

安全 | <https://docs.microsoft.com/en-us/windows-server/virtualization/hyper-v/hyper-v-on-windows-server>

Microsoft Technologies Documentation Resources Search

Windows IT Center Explore Docs Downloads Scripts Support

Docs / Windows Server / Virtualization

Filter

Hyper-V

- Technology Overview
- What's new in Hyper-V
- System requirements
- Supported Windows guest operating systems
- Supported Linux and FreeBSD VMs
- Feature compatibility by generation and guest
- Get started

Hyper-V on Windows Server 2016

10/07/2016 • 2 minutes to read • Contributors

Applies To: Windows Server 2016

The Hyper-V role in Windows Server lets you create a virtualized computing environment where you can create and manage virtual machines. You can run multiple operating systems on one physical computer and isolate the operating systems from each other. With this technology, you can improve the efficiency of your computing resources and free up your hardware resources.

See the topics in the following table to learn more about Hyper-V on Windows Server 2016.

Comments Edit Share Theme Light

In this article

- Hyper-V resources for IT Pros
- Related technologies

Is this page helpful? ×

Hardware Assisted Virtualization

Definition

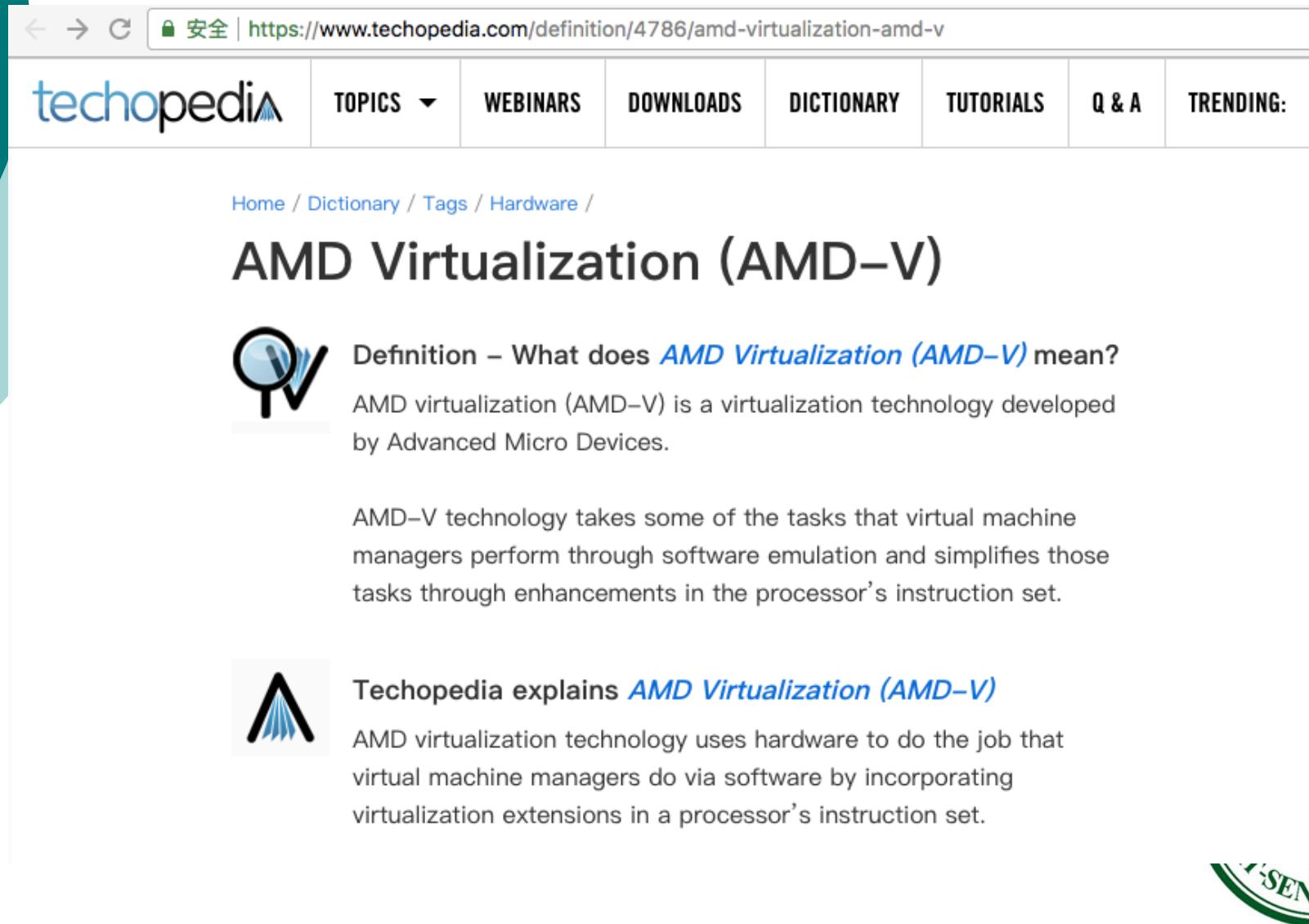
In computing, **hardware-assisted virtualization** is a platform virtualization approach that enables **efficient full virtualization** using help from **hardware capabilities**, primarily from the host processors.

- 不是独立的虚拟化技术
- 结合到全/半虚拟化技术中
- 技术要点：
 - 通过对部分全虚拟化和半虚拟化使用到的**软件技术**进行**硬件化**来提高性能
- 知名产品
 - Intel VT-x
 - AMD-V (AMD SVM)



Hardware Assisted Virtualization

安全 | <https://www.techopedia.com/definition/4786/amd-virtualization-amd-v>



The screenshot shows the Techopedia website with the URL https://www.techopedia.com/definition/4786/amd-virtualization-amd-v. The page title is "AMD Virtualization (AMD-V)". The content includes a magnifying glass icon, the definition text, and a summary of the technology's purpose. There is also a sidebar with the Techopedia logo and navigation links.

Definition – What does *AMD Virtualization (AMD-V)* mean?

AMD virtualization (AMD-V) is a virtualization technology developed by Advanced Micro Devices.

AMD-V technology takes some of the tasks that virtual machine managers perform through software emulation and simplifies those tasks through enhancements in the processor's instruction set.

Techopedia explains *AMD Virtualization (AMD-V)*

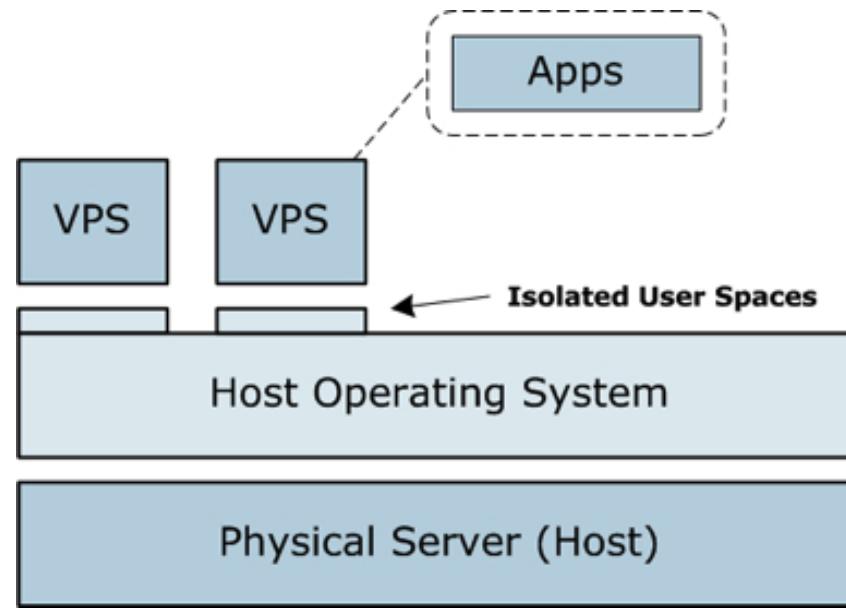
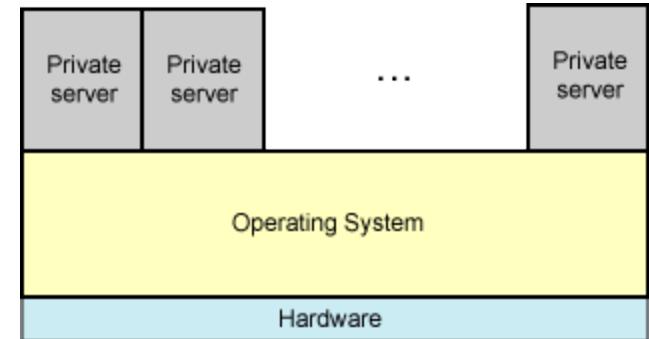
AMD virtualization technology uses hardware to do the job that virtual machine managers do via software by incorporating virtualization extensions in a processor's instruction set.



Operating System Level Virtualization

OS-level virtualization refers to an abstraction layer between traditional OS and user applications.

- 主要用于Virtual Private Server
- 技术要点：
 - 对服务器操作系统进行隔离实现虚拟化
(只有一份操作系统)
- 知名产品
 - chroot for Unix like
 - Solaris Zone
 - FreeBSD Jail
 - Linux-VServer



Operating System Level Virtualization

linux-vserver.org/Welcome_to_Linux-VServer.org

 Log in / create account

page discussion view source history

Welcome to Linux–VServer.org

English • Deutsch • español • français •

Linux–VServer provides virtualization for GNU/Linux systems. This is accomplished by kernel level isolation. It allows to run multiple virtual units at once. Those units are sufficiently isolated to guarantee the required security, but utilize available resources efficiently, as they run on the same kernel.

This site contains information relating to the use and development of virtual servers based on Linux–VServer. This particular virtual server model is implemented through a combination of "security contexts", segmented routing, chroot, extended quotas and some other standard tools.

Note: If this isn't what you are looking for, maybe [Linux Virtual Server](#) is.

Experimental Patches

Linux–VServer branch	2.3	2.3 + grsecurity
Linux kernel	Experimental	Experimental
3.18.55	vs2.3.7.5	
3.14.52	vs2.3.6.15	
3.10.104	vs2.3.6.9	

Comparisons

	硬件仿真虚拟化	全虚拟化	半虚拟化	硬件辅助虚拟化	操作系统级虚拟化
速度	<30%	30%~80%+	80%+	80%+	80%
模式	Hosted	Hosted/Bare-metal	Bare-metal	Hosted/Bare-metal	类Bare-metal
优点	Guest OS无需修改 非常适合硬件、固件及OS的开发	Guest OS无需修改，速度和功能都不错，使用非常简单	比全虚拟化架构更精简，速度上有优势	速度快	成本低效率高
缺点	速度非常慢 (有时速度比物理情况慢100倍以上)	基于Hosted模式时性能较差，特别是I/O方面	需要对Guest OS进行修改，用户体验较差	硬件实现不够优化	特定的OS
趋势	颓势但仍存	主流	一定份额	普遍采用	特定应用(VPS)



CPU虚拟化

- X86 操作系统是设计为直接运行在裸硬件设备上的，因此它们自动认为它们完全占有计算机硬件。x86 架构提供四个特权级别给操作系统和应用程序来访问硬件。
- Ring 是指 CPU 的运行级别，Ring 0 是最高级别，Ring1 次之，Ring2 更次之.....
- 就 Linux+x86 来说，操作系统需要直接访问硬件和内存，因此它的代码需要运行在最高运行级别 Ring0 上，这样它可以用特权指令，控制中断、修改页表、访问设备等等。



CPU虚拟化

- 应用程序的代码运行在最低运行级别上ring3上，不能做受控操作。
- 如果要做，比如要访问磁盘，写文件，那就要通过执行系统调用，执行系统调用的时候，CPU的运行级别会发生从ring3到ring0的切换，并跳转到系统调用对应的内核代码位置执行，这样内核就完成了设备访问，完成之后再从ring0返回ring3。
- 这个过程也称作用户态和内核态的切换。



CPU虚拟化

那么，虚拟化在这里就遇到了一个难题，因为宿主操作系统是工作在 ring0 的，客户操作系统就不能也在 ring0 了，但是它不知道这一点，以前执行什么指令，现在还是执行什么指令，但是没有执行权限是会出错的。所以这时候虚拟机管理程序（VMM）需要避免这件事情发生。

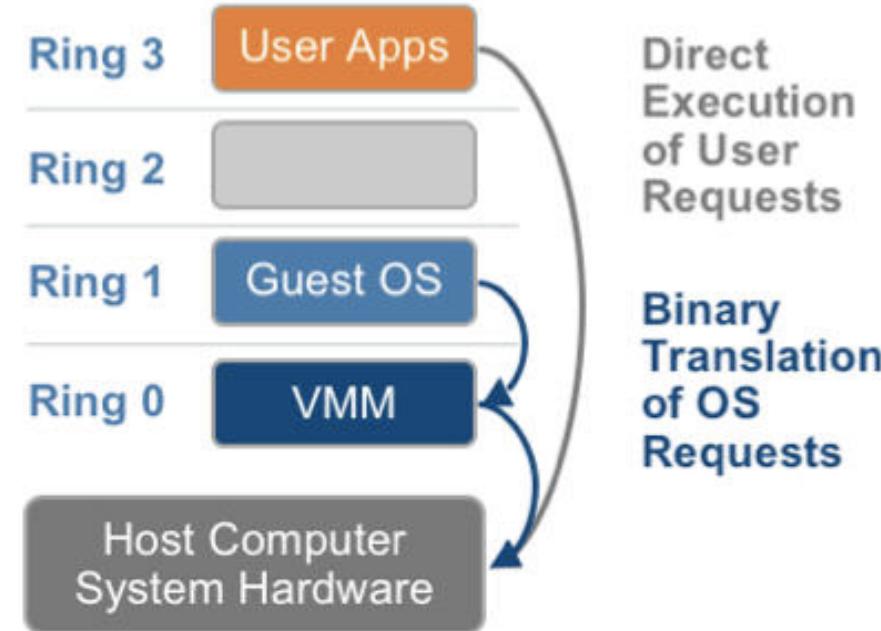
虚拟机通过VMM 实现 Guest CPU 对硬件的访问，根据其原理不同有三种实现技术：

1. 全虚拟化
2. 半虚拟化
3. 硬件辅助的虚拟化



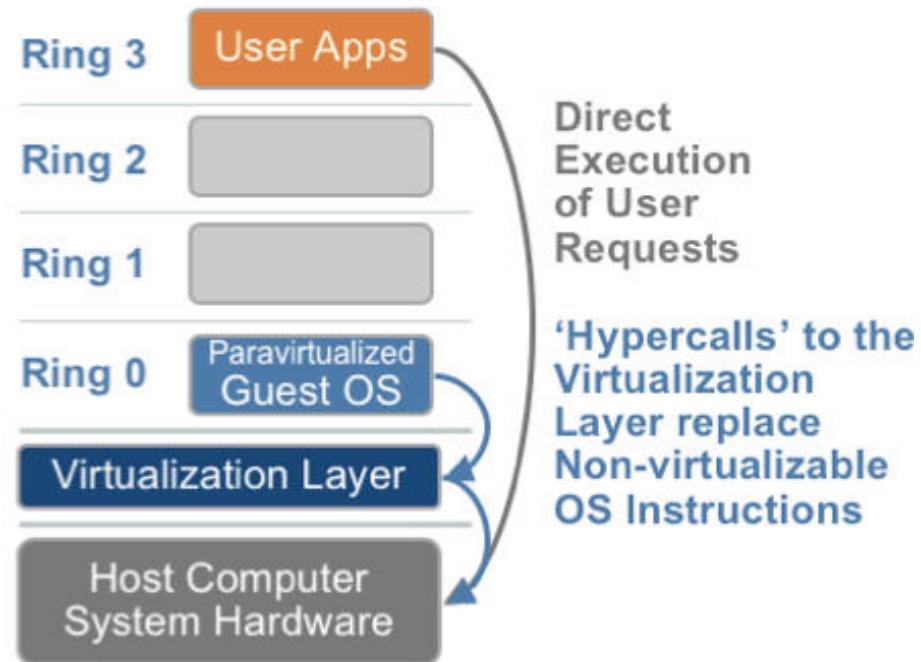
CPU全虚拟化（模拟执行）

- 客户操作系统运行在 Ring 1，它在执行特权指令时，会触发异常，然后 VMM 捕获这个异常，在异常里面做翻译，模拟，最后返回到客户操作系统内，客户操作系统认为自己的特权指令工作正常，继续运行。



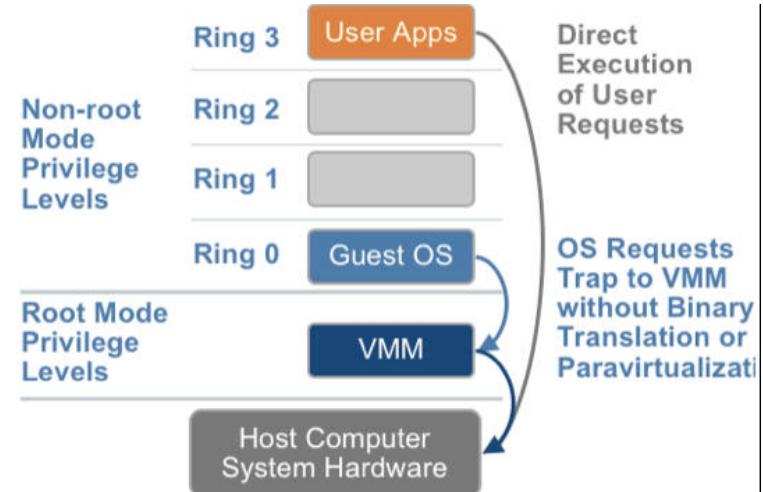
CPU半虚拟化（操作系统辅助）

- 半虚拟化的思想就是，修改操作系统内核，替换掉不能虚拟化的指令，通过超级调用（**hypcall**）直接和底层的虚拟化层**hypervisor**来通讯，**hypervisor**同时也提供了超级调用接口来满足其他关键内核操作。



CPU硬件辅助虚拟化

- 通过引入新的指令和运行模式，来让VMM和Guest OS能分别运行在其合适的模式下
- VT-x支持两种处理器工作方式
 - Root模式（Operation）：VMM运行于此模式，用于处理特殊指令
 - Non-Root模式（Operation）：Guest OS运行于此模式



CPU虚拟化

	利用二进制翻译的全虚拟化	硬件辅助虚拟化	操作系统协助/半虚拟化
实现技术	BT 和直接执行	遇到特权指令转到 root 模式执行	Hypercall
客户操作系统修改/兼容性	无需修改客户操作系统，最佳兼容性	无需修改客户操作系统，最佳兼容性	客户操作系统需要修改来支持 hypercall，因此它不能运行在物理硬件本身或其他的 hypervisor 上，兼容性差，不支持 Windows
性能	差	全虚拟化下，CPU 需要在两种模式之间切换，带来性能开销；但是，其性能在逐渐逼近半虚拟化。	好。半虚拟化下 CPU 性能开销几乎为 0，虚机的性能接近于物理机。
应用厂商	VMware Workstation/QE MU/Virtual PC	VMware ESXi/Microsoft Hyper-V/Xen 3.0/KVM	Xen



内存虚拟化

- 目标：
 - 做好虚拟机内存空间之间的隔离，使每个虚拟机都认为自己拥有了整个内存地址，**并且效率也能接近物理机。**
- 全虚拟化：影子页表 **Shadow Page Table**
 - 为每个**Guest**都维护一个影子页表，写入虚拟化之后的内存地址映射关系，而**Guest OS**的页表则无需变动
 - VMM将影子页表交给**MMU**进行地址转换
- 半虚拟化：页表写入法
 - 当**Guest OS**创建一个新的页表时，其会向**VMM**注册该页表
 - 之后在**Guest**运行的时候，**VMM**将不断地管理和维护这个表，使**Guest**上面的程序能直接访问到合适的地址
- 硬件辅助虚拟化：扩展页表**EPT** (**Extended Page Table**)
 - EPT通过使用硬件技术，在原有的页表的基础上，增加了一个**EPT页表**
 - 通过这个页表能够将**Guest**的物理地址直接翻译为主机的物理地址



I/O虚拟化

○ 目标：

- 不仅让虚拟机访问到它们所需要的I/O资源，而且要做好它们之间的隔离工作，更重要的是，减轻由于虚拟化所带来的开销。

○ 全虚拟化：

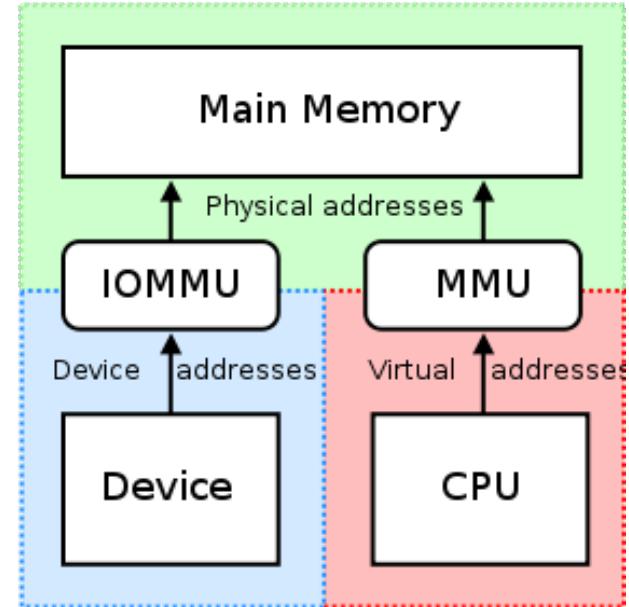
- 通过模拟I/O设备（磁盘和网卡等）来实现虚拟化。
- Guest OS每次I/O操作都会陷入到VMM，让VMM来执行。
- 对Guest OS而言，它看到就是一组统一的I/O设备，完全透明。

○ 半虚拟化：

- 通过前端（Front-End）/后端（Back-End）架构，将Guest的I/O请求传递到特权域（Privileged Domain，也被称为Domain-0）。



I/O虚拟化



○ 软件模拟虚拟化：

- 用软件模拟I/O设备
- Guest OS的操作被VMM捕获并交给Host OS的用户态进程，由其进行系统调用

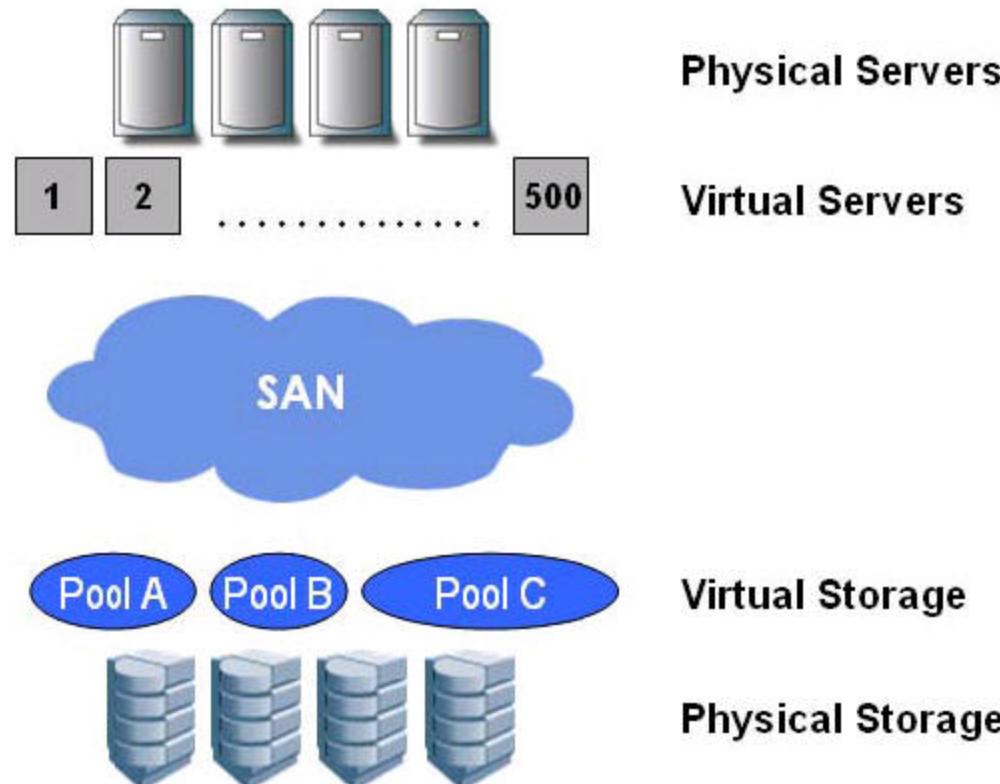
○ 硬件辅助虚拟化（直接划分）：IOMMU

- 代表产品：Intel的VT-d, AMD-Vi, PCI-SIG的IOV
- 其核心思想就是让虚拟机能直接使用物理设备
- 技术要点：I/O地址访问和DMA的问题
 - 通过采用DMA重映射（Remapping）和I/O页表来解决



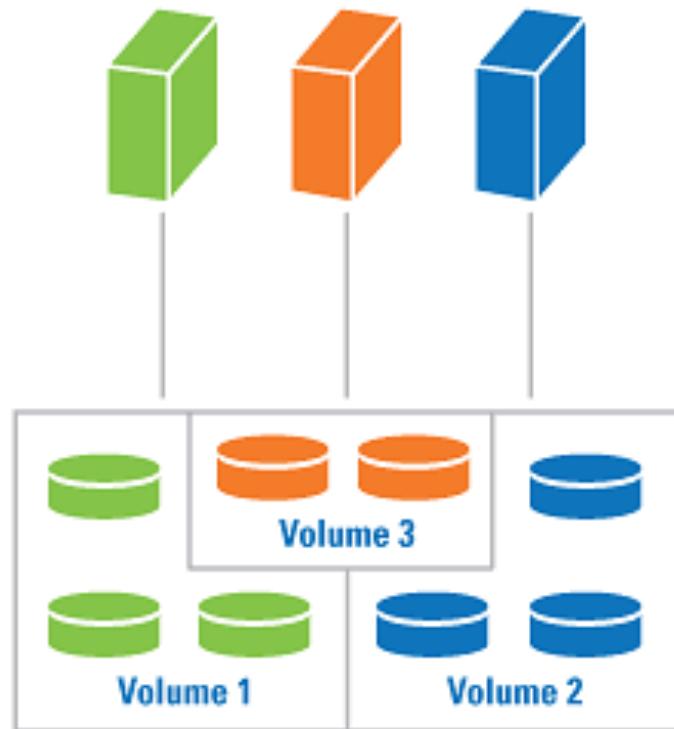
存储虚拟化

- 目标：实现与应用/网络无关的数据存储和管理
- 存储功能 → ||| 抽象隔离 ||| ← 应用、主机、网络

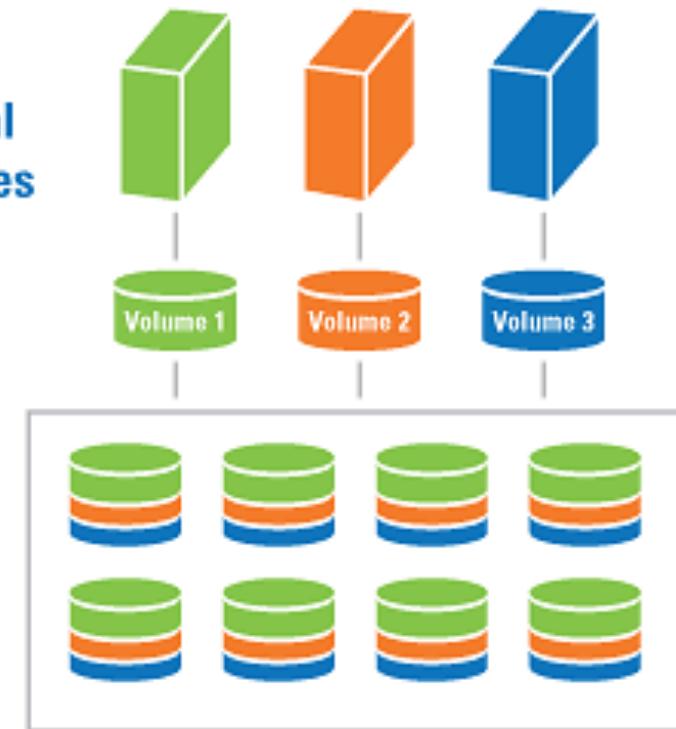


传统存储 vs. 虚拟存储

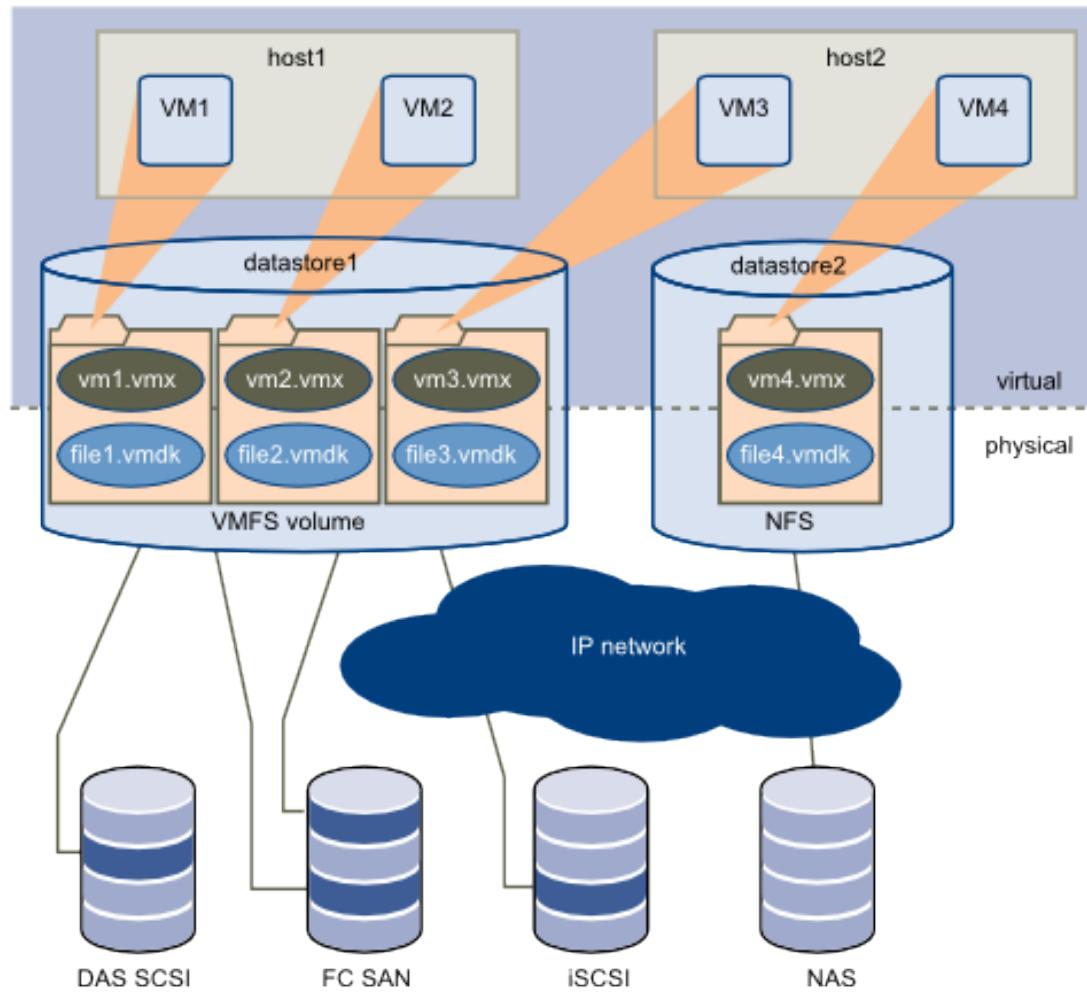
Traditional Disk Mapping



Virtualized Storage Disk Mapping

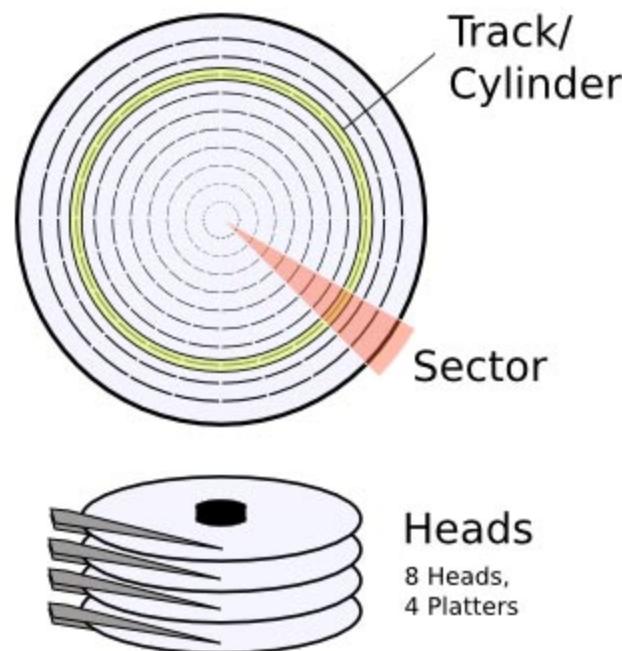


整体架构（示例）



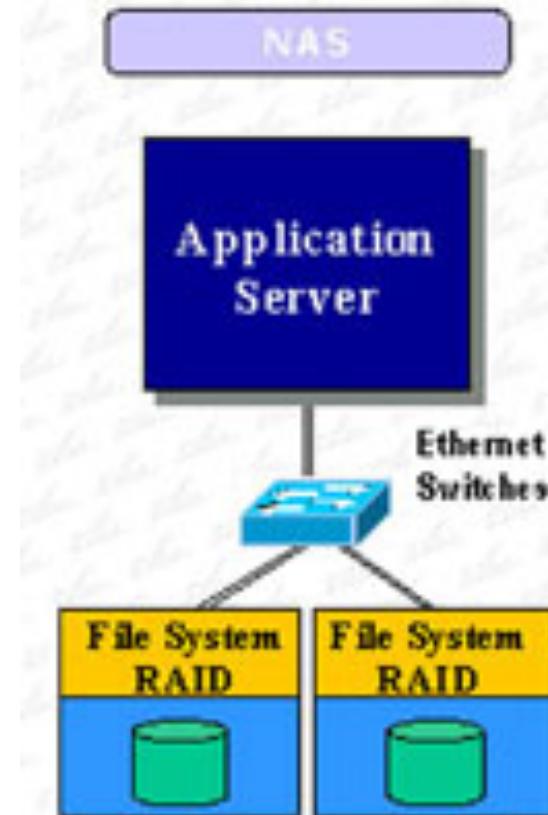
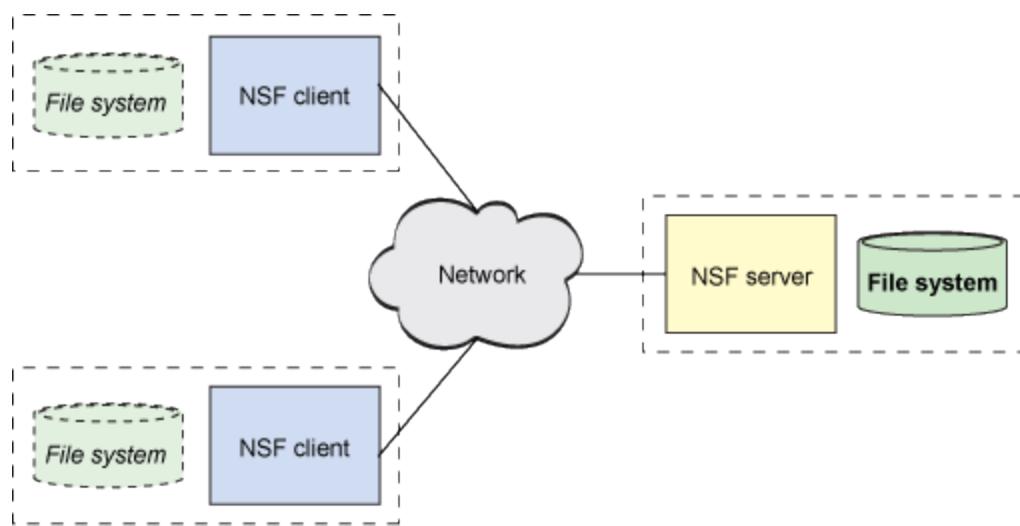
设备层虚拟化

- 目标：
 - 虚拟化磁盘的物理属性，实现统一的寻址
(不同类型的物理磁盘可以有相同的访问方式)
- 实现技术：
 - 磁盘固件，CHS信息→逻辑块编号



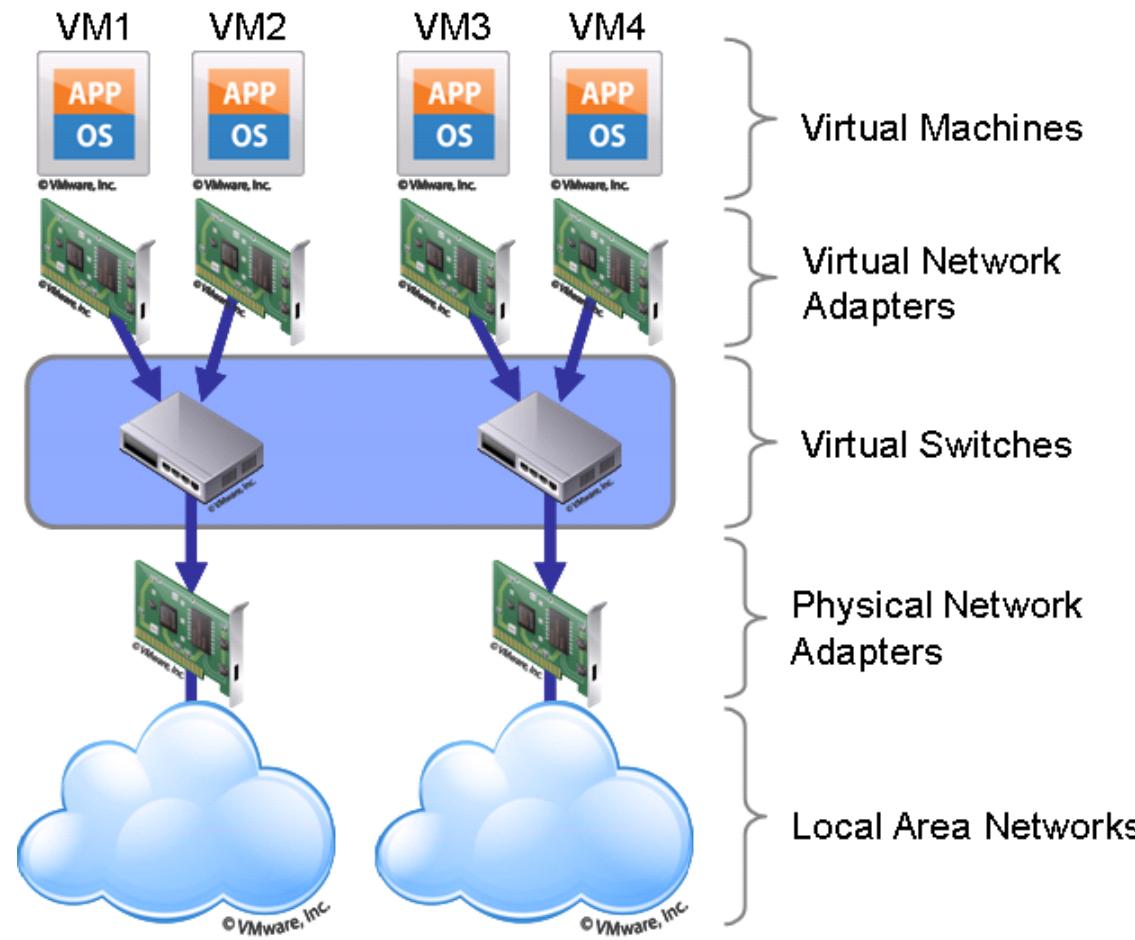
文件/记录层虚拟化

- 目标：
 - 整合不同的文件系统
 - 使上层用户能够透明、高效地访问存储在远程的文件。
- 实现技术
 - NFS、CIFS、NAS
 - 分布式文件系统



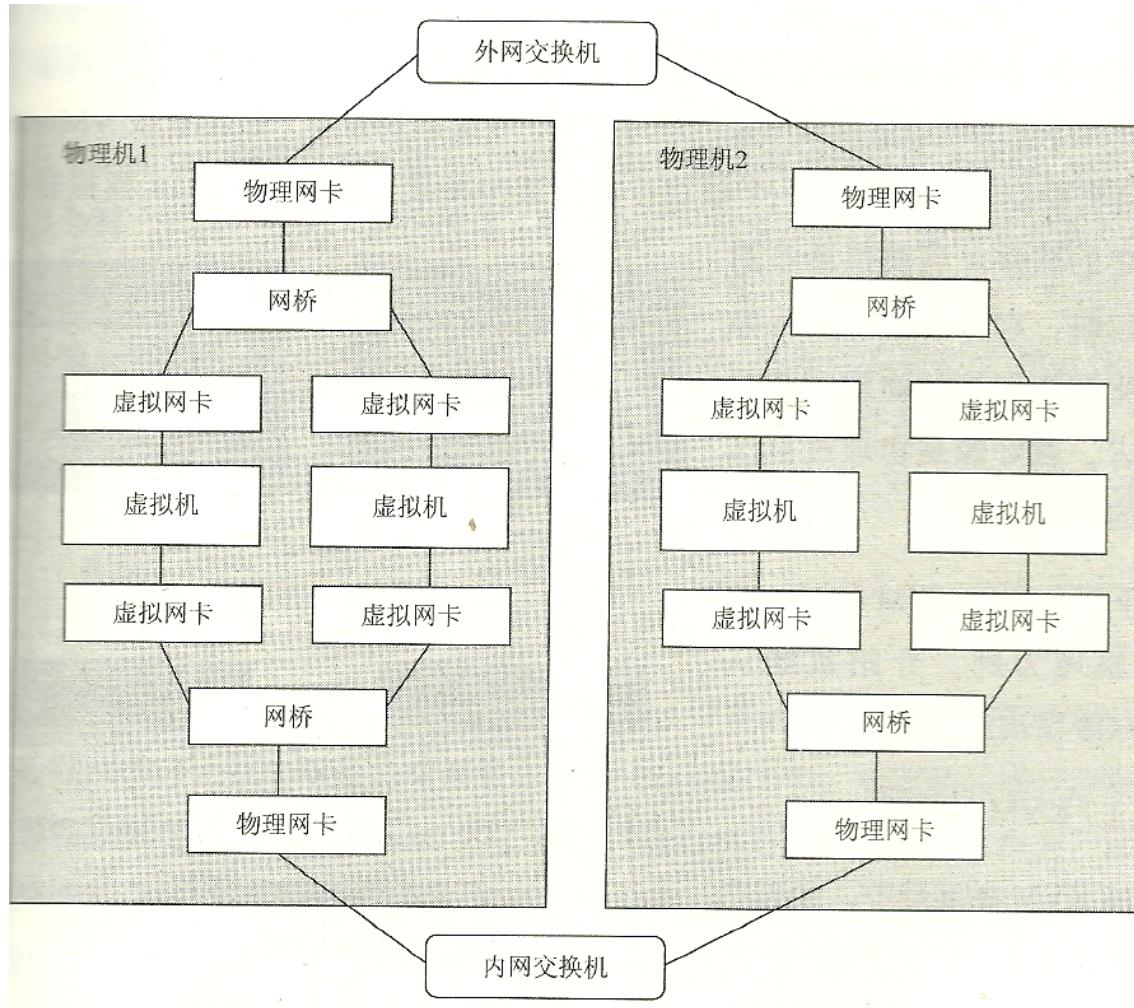
网络虚拟化

- 将网络资源和功能集成到一个软件中统一管控



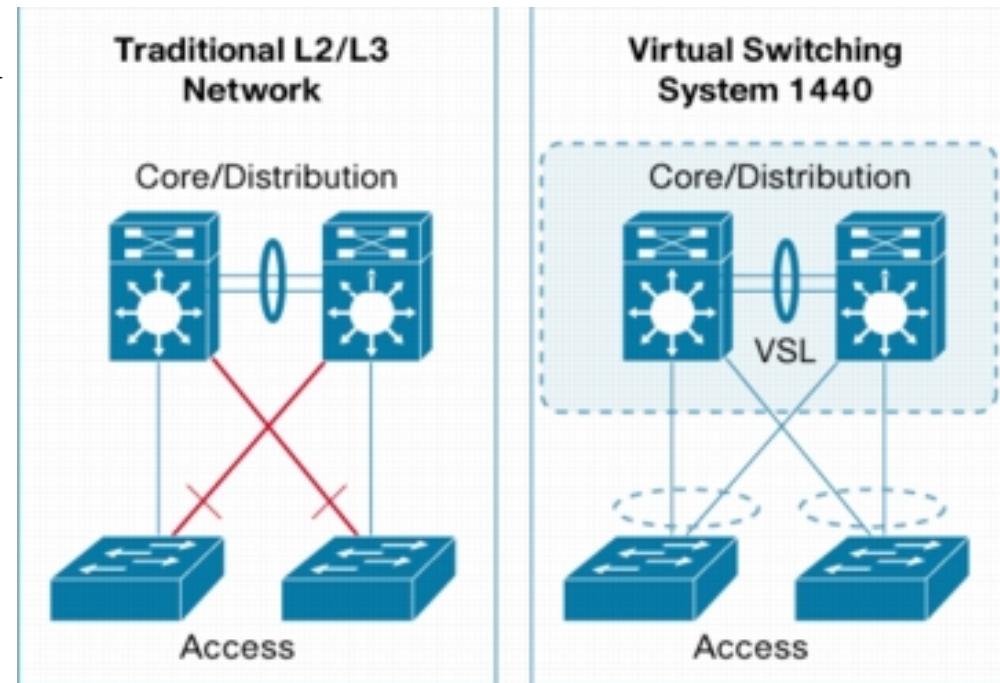
主机网络虚拟化

- 单台物理主机的网络设备的虚拟化

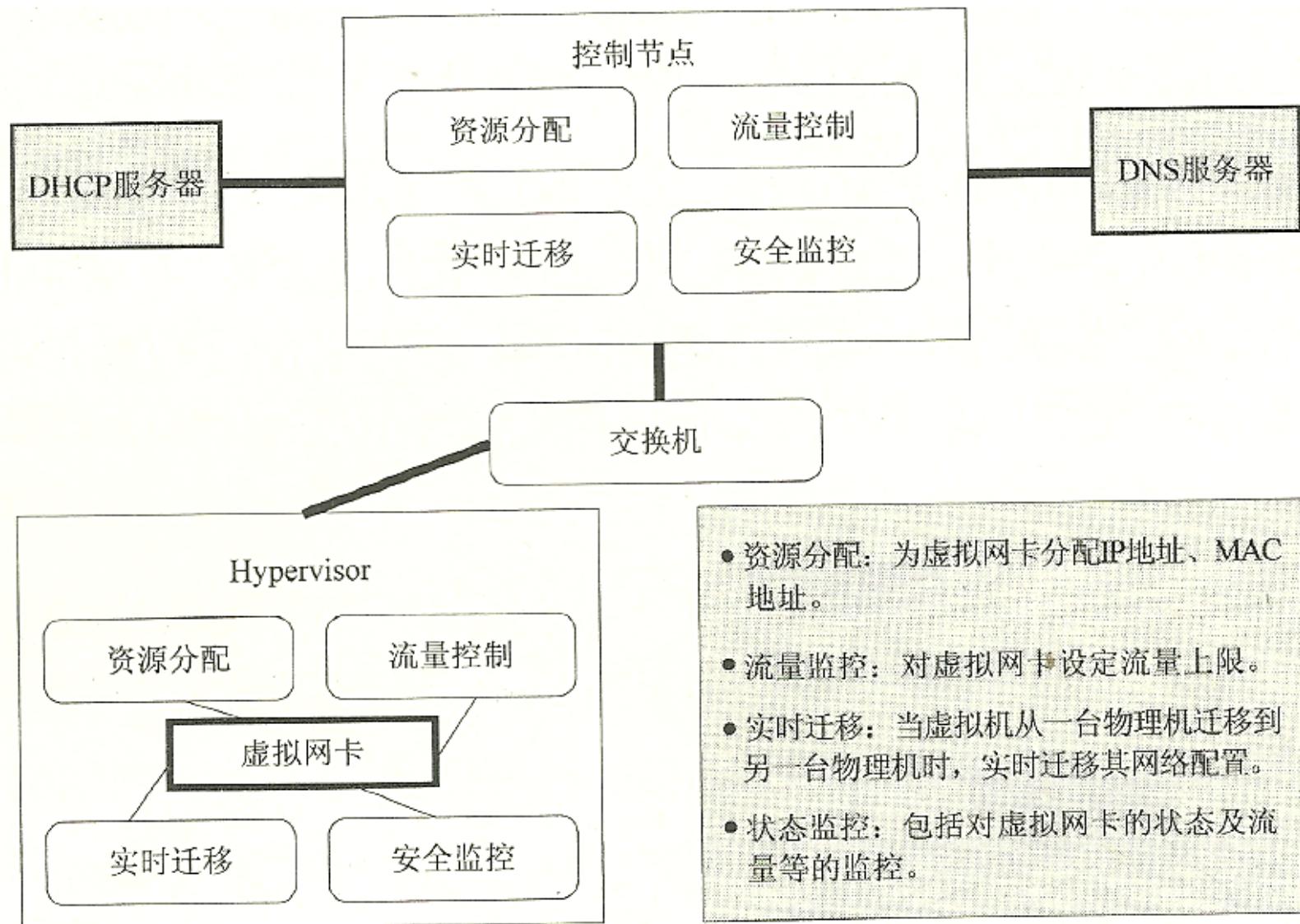


网络设备虚拟化

- 适应云计算环境的需求
 - 增加容量、应对横向流量、大缓存低延迟等。
- 技术要点
 - 网络交换设备层面实施虚拟化
- 两种形式
 - 横向整合：多个变一个
 - 纵向分割：VLAN等



网络虚拟化管理

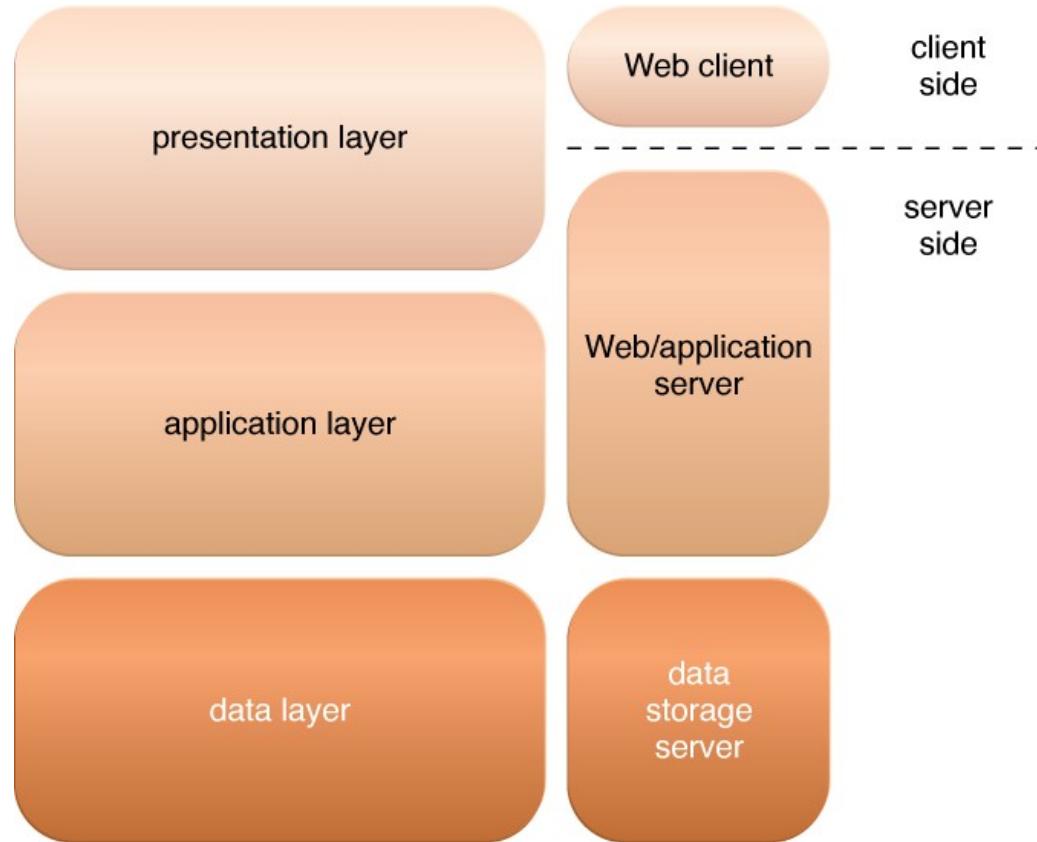


§5.4 Web技术

- 用于访问云服务
 - I、P、S
- 需求要点
 - 迅速的动态交互
 - 丰富的交互手段
 - 高效的呈现性能
 - 界面可定制
 - 离线使用
 - 直观教程



Web应用



Copyright © Arcitura Education

Figure 5.10 – Web应用的三层基本架构



基于插件的Web呈现技术

- Flash
- Silverlight
- JavaFX



基于浏览器的Web呈现技术

- **HTML5**

- 实现网页结构与内容描述的扩展
- 满足功能需求
- 满足离线使用需求

- **CSS3**

- 页面显示特效
- 更接近客户端效果

- **Ajax**

- 部分、异步交互数据
- 避免页面重载导致的不连贯



HTML5

- W3C, 2007
- 主要改进（相比HTML4及以前版本）
 - 增加**Audio**、**Video**等多媒体元素
 - 嵌入编解码器
 - 支持定时播放、播放控制
 - 支持H.264和Ogg
 - 增加**Canvas**元素
 - 用JavaScript绘图
 - 提供**Geolocation**地理位置API
 - 本地存储功能
 - 增加了结构化标签
 - . . .



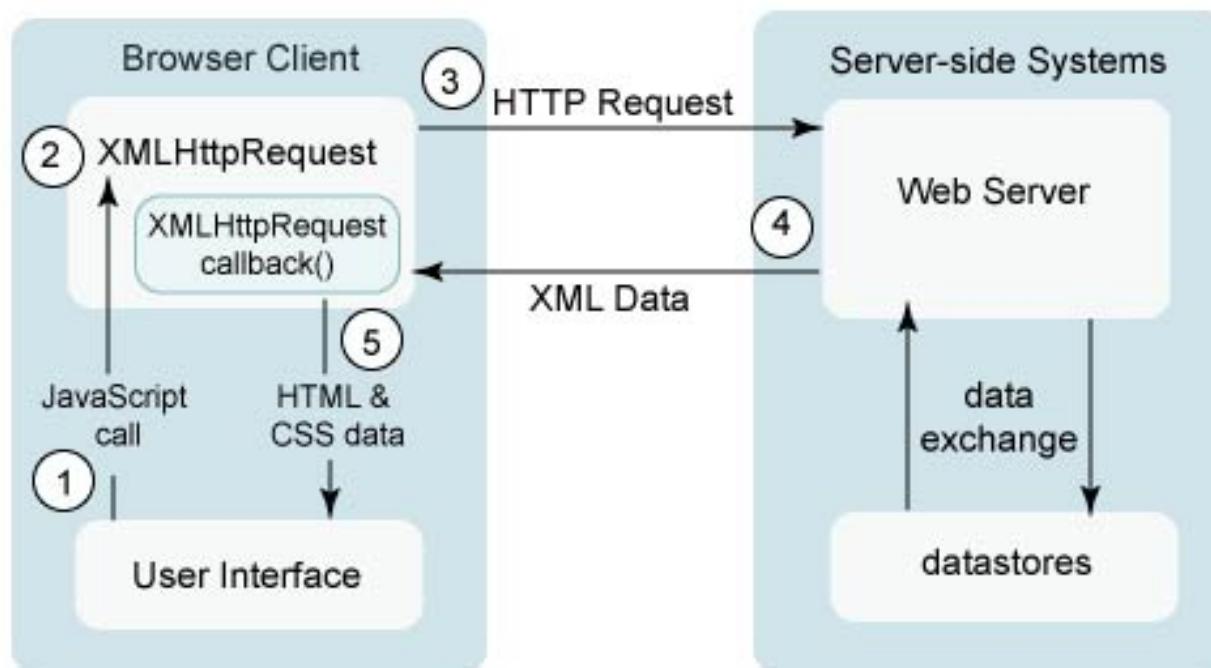
CSS3

- Cascading Style Sheets, 层叠样式表
- 用来为结构化文档（如HTML文档或XML应用）添加样式（字体、间距和颜色等）
- CSS3的新特性
 - 文字特效
 - 动画支持
 - . . .



Ajax

- 能在不更新整个页面的前提下维护数据
- 不是一种单一的技术，而是利用了一系列技术



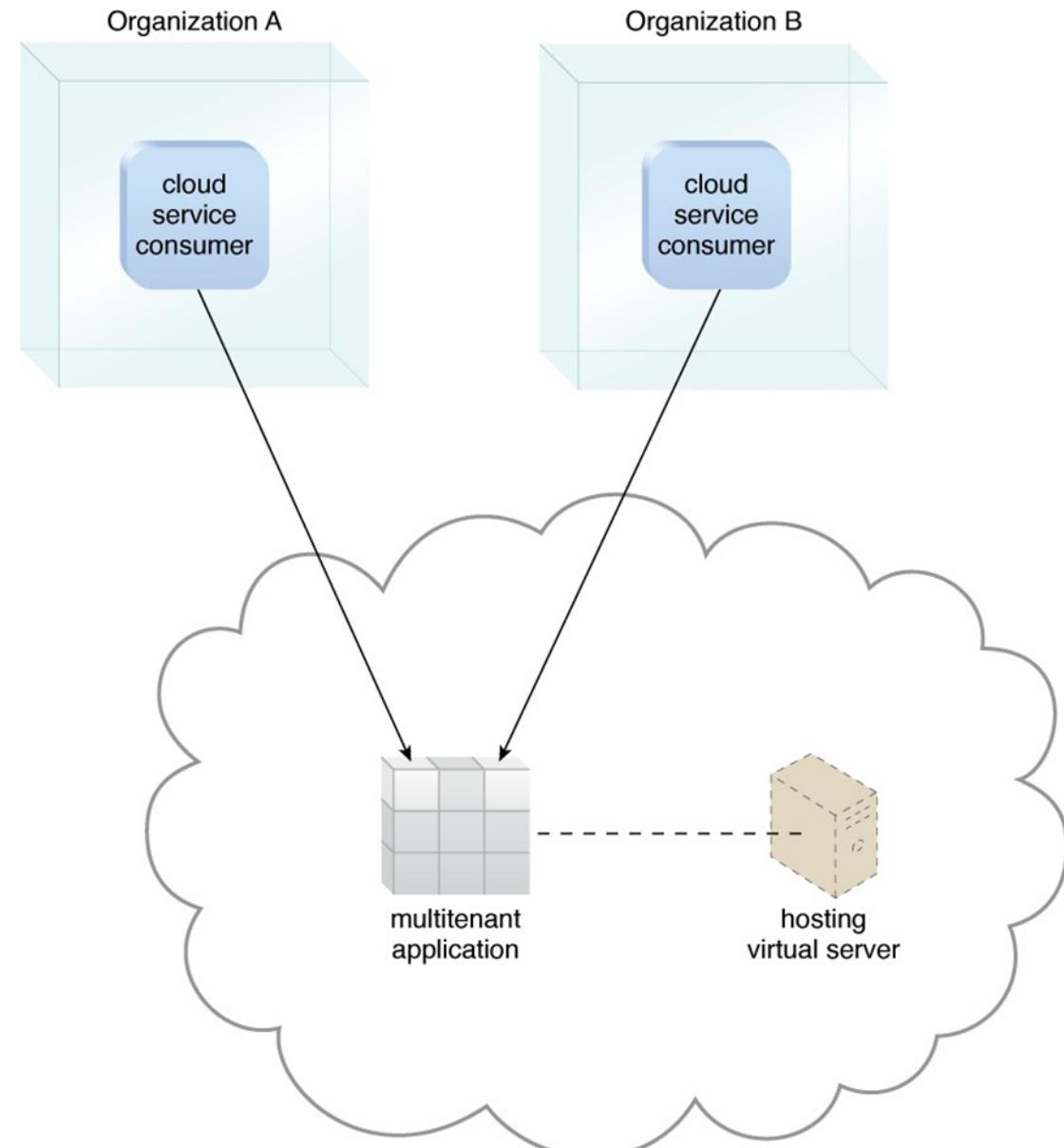
§5.5 多租户技术

- 设计多租户应用的目的是使多个用户（租户）在逻辑上同时访问同一个应用
- 多租户应用架构通常比单租户应用要复杂的多
- 多租户应用需要支持多用户对各种构件的共享（包括入口、数据模式、中间件和数据库），同时还需要保持安全等级来隔离不同租户的操作环境。



多租户技术

Figure 5.11 – 同时为
多个云服务用户提供
服务的多租户应用



多租户技术

- 每个租户可以独立定制以下应用特性：
 - 用户界面 (**user interface**) —— 租户可以定义具有专门界面外观的应用接口。
 - 业务流程 (**business process**) —— 在实现应用时，租户可以定制业务处理的规则、逻辑和工作流。
 - 数据模型 (**data model**) —— 租户可以扩展应用的数据模式，以包含、排除或者重命名应用数据结构的字段。
 - 访问控制 (**access control**) —— 租户可以独立控制用户或群组的访问权限。

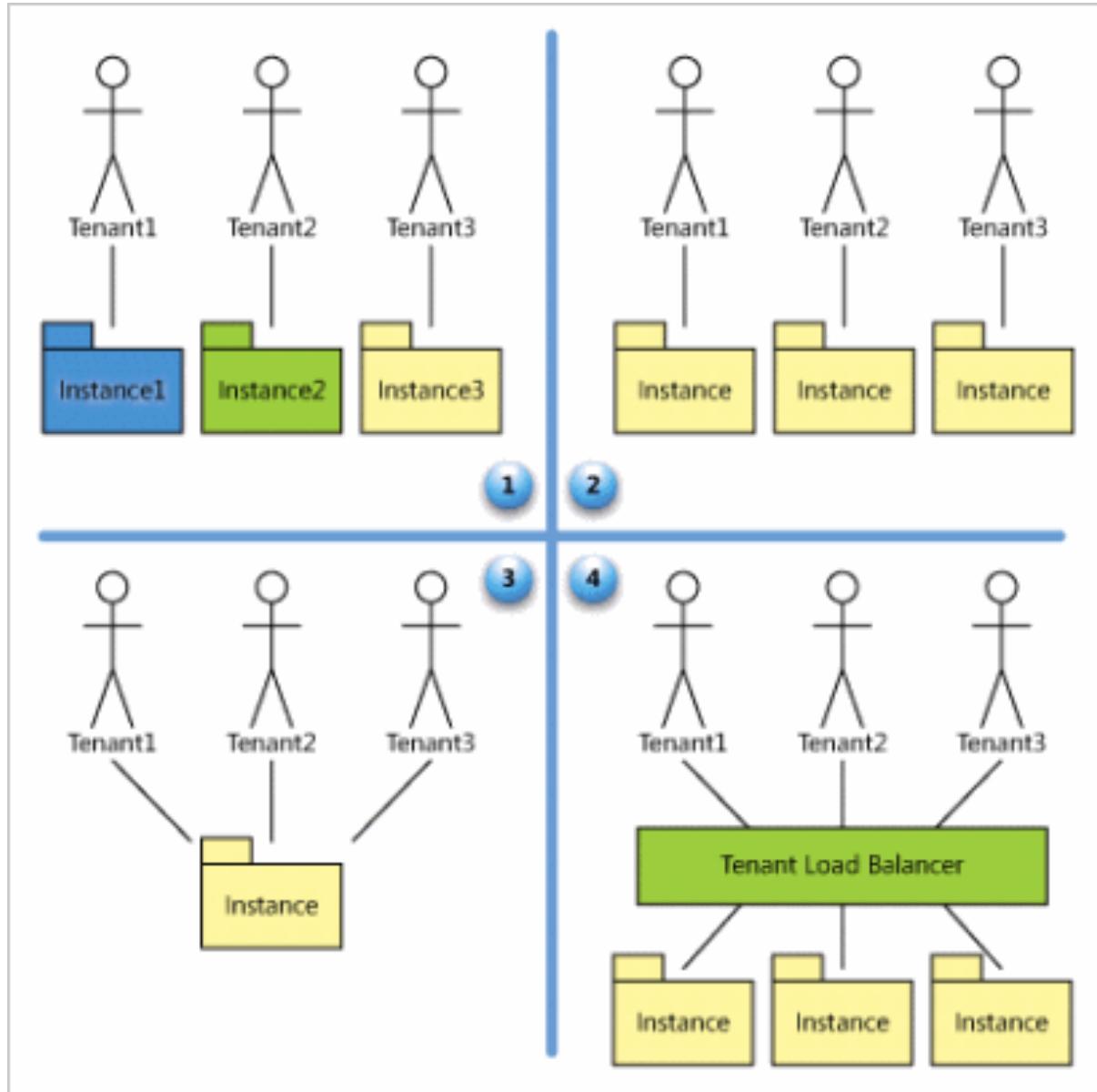


多租户技术

- 多租户应用的一般特点包括：
 - 使用隔离（**usage isolation**）——一个租户的使用行为不会影响到该应用对其他租户的可用性和性能。
 - 数据安全（**data security**）——租户不能访问其他租户的数据。
 - 可恢复性（**recovery**）——每个租户的数据备份和恢复过程都是分别执行的。
 - 应用升级（**application upgrade**）——共享软件构件的同步升级不会对租户造成负面影响。
 - 可扩展性（**scalability**）——根据现有租户增长的使用需求或租户数量的增加来扩展应用。
 - 使用计费（**metered usage**）——根据租户实际使用的应用处理和功能来收费。
 - 数据层隔离（**data tier isolation**）——租户拥有独立的且与其他租户隔离的数据库、表格和模式。



成熟度及分类



成熟度及分类

○ I型：特定/定制 Ad-hoc / custom

- 即传统的“软件运营”
- 用户都有定制的程序版本，运行自己的实例
- 传统软件少量改变即可移植
- 通过合并服务器、系统维护降低成本

○ II型：可配置 Configurable

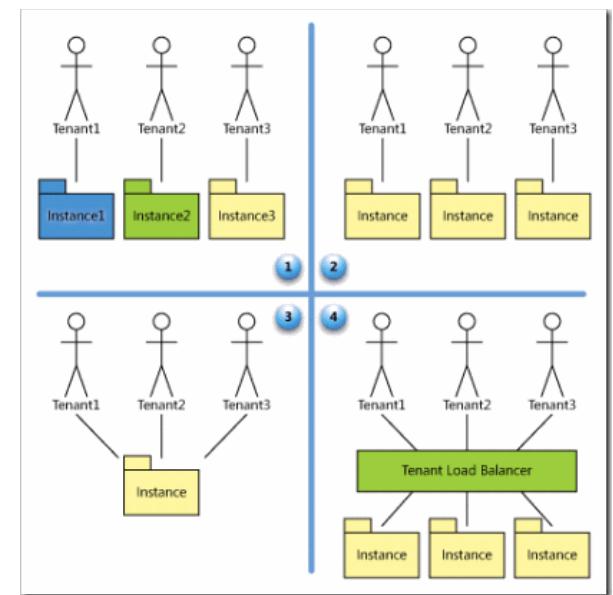
- 详细的配置选项对元数据进行配置
- 不同用户使用同一程序的不同实例

○ III型：多租户 Multi-tenancy

- 不同用户使用同一程序的单一实例
- 隔离用户的操作界面和应用数据

○ IV型：可扩展性 Scalable

- 通过多层架构对服务器进行动态调整适应可变负载



多租户与虚拟化

- 虚拟化：一个物理服务器上可以容纳**服务器**环境的多个虚拟副本。每个副本都可以提供给不同的用户，可以独立配置，还可以包含自己的操作系统和应用程序。--IaaS
- 多租户：一个物理或虚拟服务器运行着一个**应用程序**，该应用程序允许被多个不同的用户共享。每个用户都感觉只有自己在使用该应用程序。--SaaS



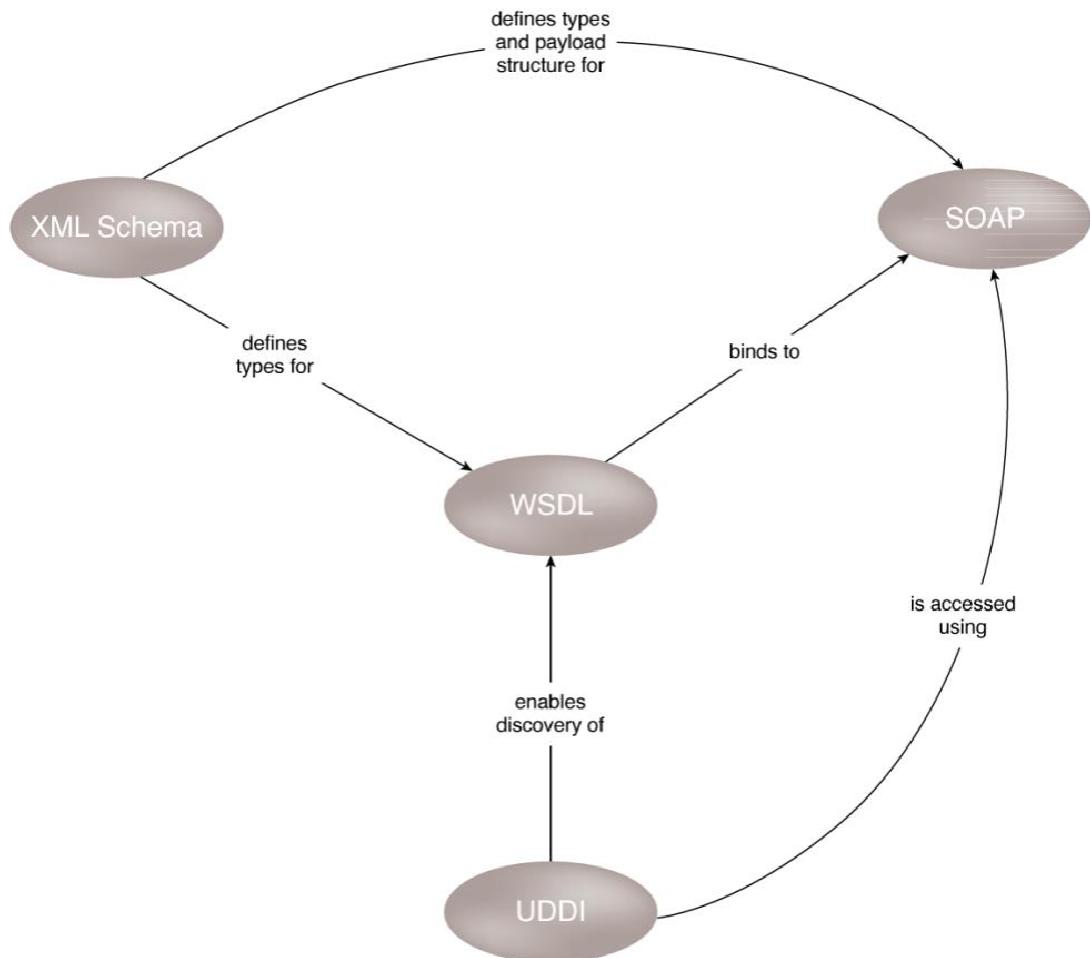
§5.6服务技术

- 与**XML**一起，**Web**服务的核心技术表现为如下工业标准：
 - **Web**服务描述语言（WSDL）
 - **XML**模式描述语言（**XML Schema**）
 - **SOAP**（简单对象访问协议）
 - 统一描述、发现和集成（UDDI）
- **Note:** 上述四项技术形成了第一代**Web**服务技术（见**Figure5.12**）。第二代**Web**服务技术（通常被称为**WS-***）的开发更加全面，可以应对各种其他功能问题，如安全性、可靠性、事务处理、路由和业务流程自动化。



Web服务

Figure 5.12 – 第一代Web服务技术相互关系示意图



Copyright © Arcitura Education



服务技术

- **REST服务**: 是按照一组约束条件设计，这组约束条件使得服务架构模拟WWW的属性。**REST有6个设计约束**，分别为：
 - 客户端-服务器
 - 无状态
 - 缓存
 - 接口/统一合约
 - 层次化系统
 - 按需编码
- **服务代理**是事件驱动程序，它在运行时拦截消息。分为主动服务代理和被动服务代理。
- **服务中间件**：服务技术的大框架下是市场巨大的中间件平台。两种常见的类型是**企业服务总线（ESB）**和**业务流程平台**。



本章小结

- 数据中心的基本使能技术
 - 管理与部署、服务提供
- 虚拟化技术
 - 技术种类
 - 虚拟化的功能
- Web技术
- 多租户技术
 - 成熟度分类
- 服务技术



课后题

- 1、分析讨论本章介绍的相关的云使能技术。
- 2、分析讨论数据中心的两种存储硬件，以及相关的存储系统技术和增加存储容量的技术。
- 3、区别多租户和虚拟化实例这两个概念。

