

第七讲 云存储之 数据中心虚拟化

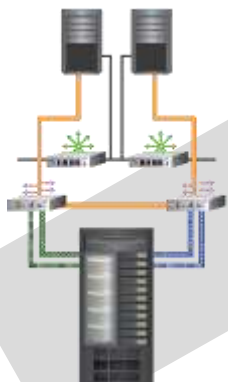


计算虚拟化

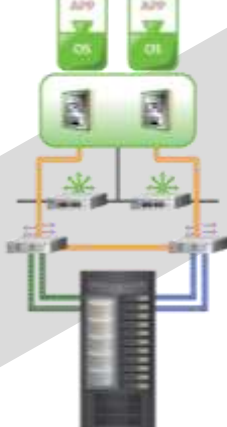
虚拟化数据中心

将传统数据中心 (CDC) 转变为虚拟化数据中心 (VDC) 需要对数据中心的
核心要素进行虚拟化。

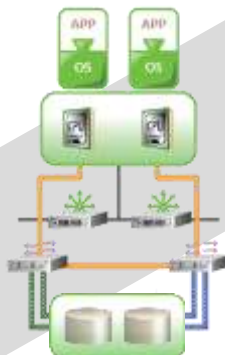
传统数据中心 (CDC)



虚拟化计算



虚拟化存储



虚拟化网络



虚拟化数据中心 (VDC)



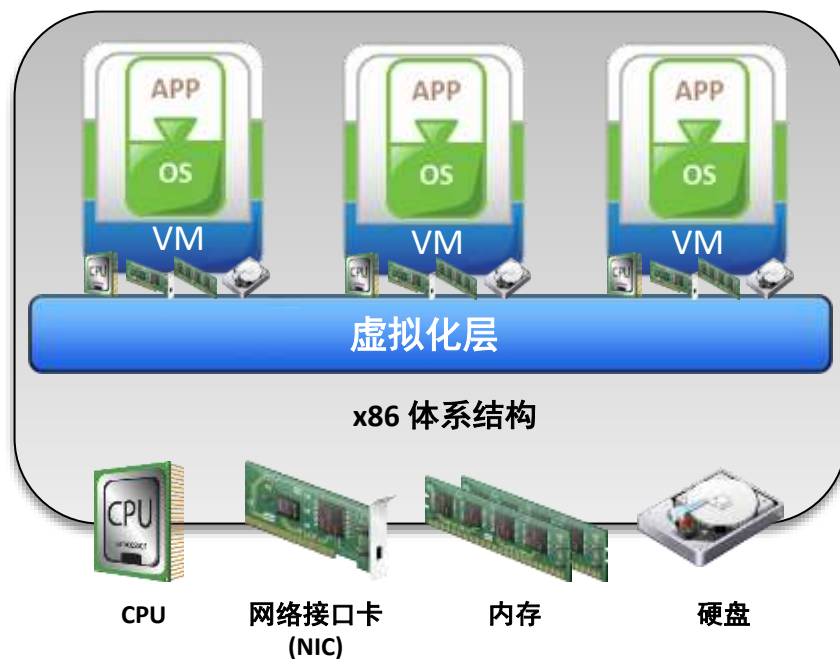
采用分阶段的方式实施虚拟化基础架构有利于实现更平稳的核心要素虚拟化转移。

计算虚拟化

计算虚拟化

它是一种用来掩蔽或抽象化物理计算硬件并实现在单个或群集物理机上并发运行多个操作系统 (OS) 的技术。

- 支持创建多个虚拟机 (VM)，且各自运行操作系统和应用程序
 - ▶ 虚拟机是外观和行为均与物理机相似的逻辑实体
- 虚拟化层驻留在硬件和虚拟机之间
 - ▶ 也称为“虚拟机管理程序”
- 虚拟机将获得标准化硬件资源



计算虚拟化的需求



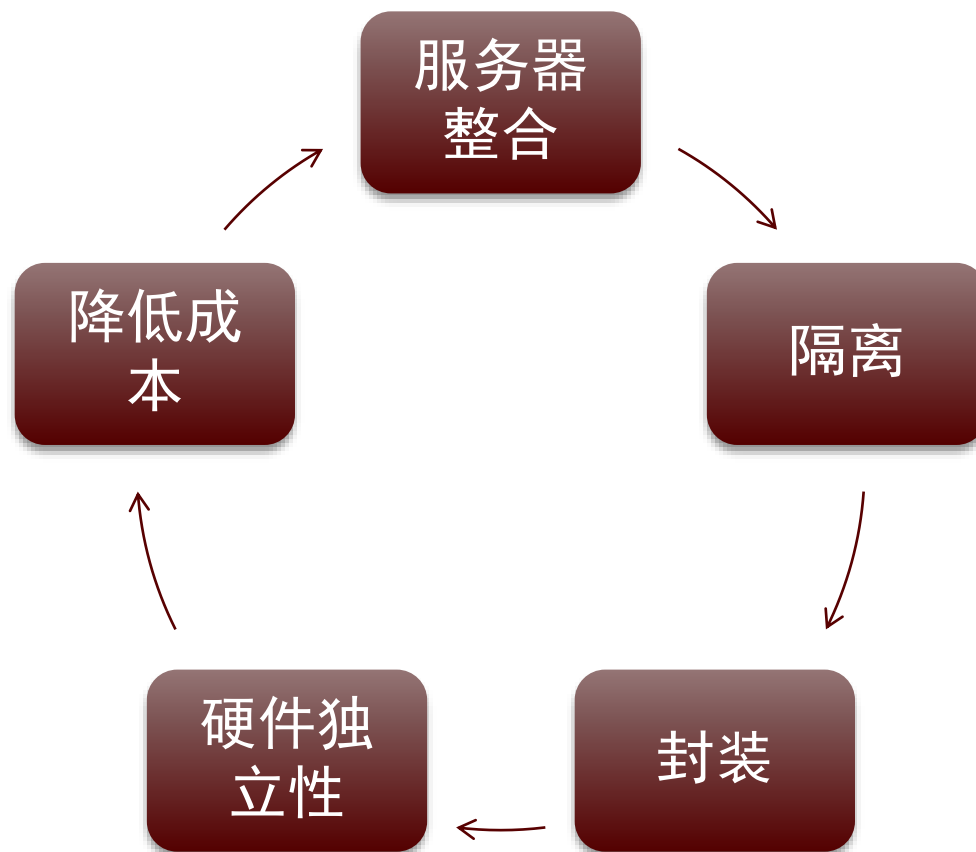
虚拟化之前

- 每台机器每次运行单个操作系统 (OS)
- 软件和硬件紧密结合
- 在同一台机器上运行多个应用程序时可能发生冲突
- 无法充分利用资源
- 昂贵而且不灵活

虚拟化之后

- 每台机器并发运行多个操作系统 (OS)
- 使得操作系统和应用程序与硬件相互独立
- 虚拟机之间相互隔离，因此不会发生冲突
- 提高资源利用率
- 以较低成本提供灵活的基础架构

计算虚拟化的优势

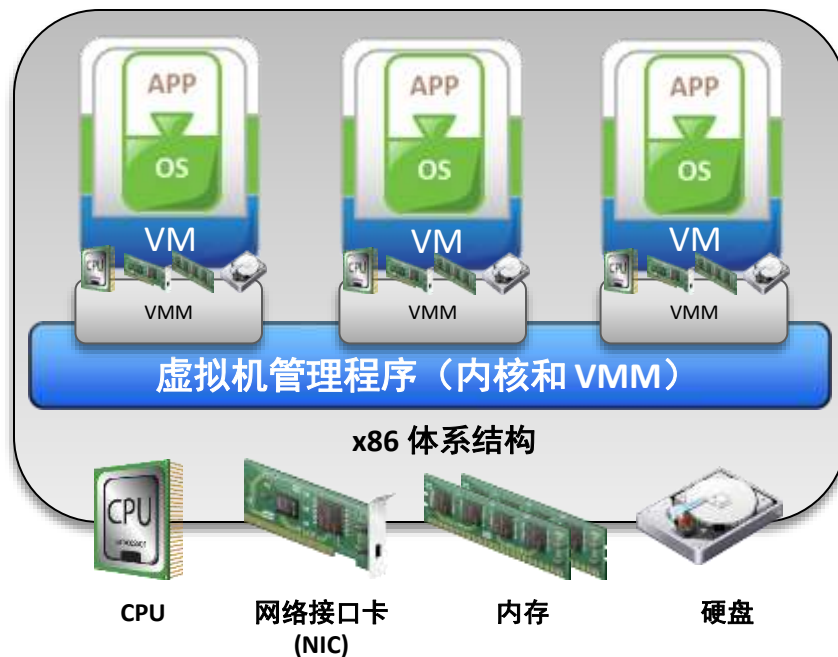


虚拟机管理程序

虚拟机管理程序

它是一种允许多个操作系统 (OS) 在一台物理机上并发运行并直接与物理硬件交互的软件。

- 包含两个组件
 - ▶ 内核
 - ▶ 虚拟机监视器 (VMM)



虚拟机管理程序的类型



第 1 类：裸机虚拟机管理程序



第 2 类：托管虚拟机管理程序

第 1 类：裸机虚拟机管理程序

- 它是一种操作系统 (OS)
- 它在 x86 裸机硬件上安装和运行
- 它需要使用经认证的硬件

第 2 类：托管虚拟机管理程序

- 它作为应用程序安装和运行
- 它依赖物理机上运行的操作系统 (OS) 来提供设备支持和物理资源管理

x86 硬件虚拟化

- 操作系统 (OS) 设计为在裸机硬件上运行, 并完全拥有硬件
 - ▶ x86 体系结构提供四个权限级别
 - ▶▶ Ring 0、1、2 和 3
 - ▶▶ 用户应用程序在 Ring 3 中运行
 - ▶▶ 操作系统在 Ring 0（最高权限）中运行
- 虚拟化 x86 硬件时面临的难题
 - ▶ 需要将虚拟化层置于操作系统层以下
 - ▶ 难以在运行时捕获和转换操作系统特权指令(包含中断处理、定时器控制和输入/输出指令的)
- 用来将计算虚拟化的技术
 - ▶ 完全虚拟化、准虚拟化和硬件辅助虚拟化



完全虚拟化

- 虚拟机监视器 (VMM) 在特权 Ring 0 中运行
- VMM 将来宾操作系统 (OS) 与底层物理硬件进行分离
- 每个虚拟机被分配一个 VMM
 - ▶ 为每个虚拟机提供虚拟组件
 - ▶ 对不可虚拟化的操作系统指令执行二进制转换 (BT)
- 来宾操作系统不会感知到被虚拟化



准虚拟化

- 通过操作系统辅助完成虚拟化
- 来宾操作系统 (OS) 知道自己正被虚拟化
- 来宾操作系统在 Ring 0 中运行
- 使用修改过的来宾 OS 内核，例如 Linux 和 OpenBSD
- 不支持未修改的来宾操作系统，例如 Microsoft Windows
- 准虚拟化技术的优点是性能高。经过准虚拟化处理的服务器可与hypervisor协同工作，其响应能力几乎不亚于未经过虚拟化处理的服务器。准虚拟化与完全虚拟化相比优点明显，以至于微软和VMware都在开发这项技术，以完善各自的产品。



硬件辅助虚拟化

- 实现方式是使用虚拟机管理程序感知的 CPU 来处理特权指令
 - ▶ 降低了完全虚拟化和准虚拟化所导致的虚拟化开销
 - ▶ 在硬件中提供 CPU 和内存虚拟化支持
- 通过 x86 处理器体系结构中的 AMD-V 和 Intel VT 技术实现



虚拟机

- 从用户的角度而言，逻辑计算系统
 - ▶ 像物理机一样运行操作系统 (OS) 和应用程序
 - ▶ 包含 CPU、RAM、磁盘和网络接口卡 (NIC) 等虚拟组件
- 从虚拟机管理程序的角度而言
 - ▶ 虚拟机 (VM) 是离散的文件集，例如配置文件、虚拟磁盘文件、虚拟 BIOS 文件、虚拟机交换文件和日志文件等

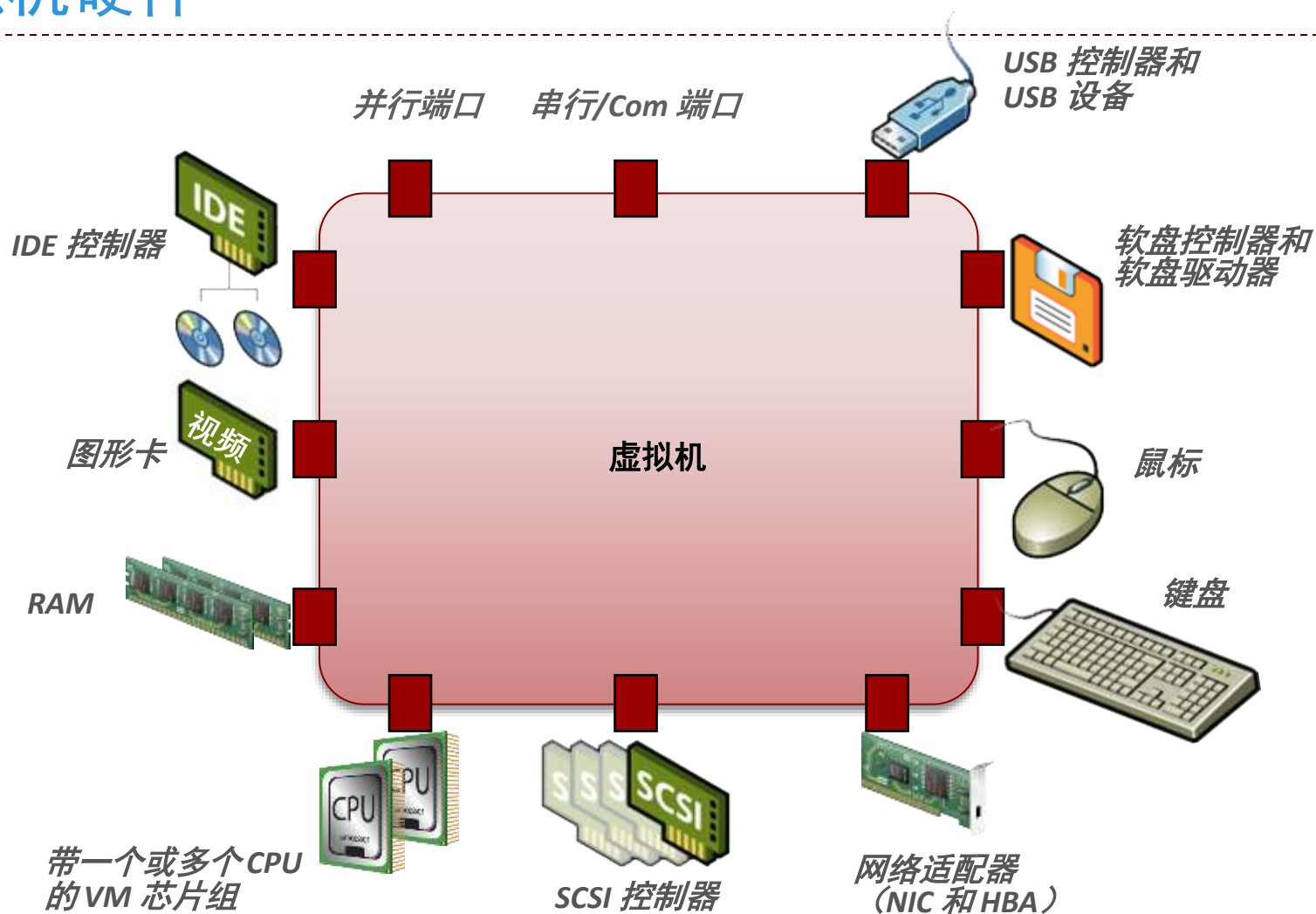


虚拟机文件

文件名	描述
虚拟 BIOS 文件	<ul style="list-style-type: none">• 存储虚拟机 (VM) 的 BIOS 状态
虚拟交换文件	<ul style="list-style-type: none">• 是虚拟机的分页文件，用于备份虚拟机 RAM 内容• 该文件仅在虚拟机运行时存在
虚拟磁盘文件	<ul style="list-style-type: none">• 存储虚拟机磁盘驱动器的内容• 对虚拟机而言类似于物理磁盘驱动器• 虚拟机可以拥有多个磁盘驱动器
日志文件	<ul style="list-style-type: none">• 保存虚拟机活动的日志记录• 适用于故障排除
虚拟配置文件	<ul style="list-style-type: none">• 存储在创建虚拟机的过程中所选的配置信息• 包括 CPU 数量、内存、网络适配器数量和类型以及磁盘类型等信息



虚拟机硬件



虚拟机硬件组件

虚拟硬件	描述
vCPU	<ul style="list-style-type: none">• 虚拟机 (VM) 可以配置一个或多个虚拟 CPU• 可以更改分配到虚拟机的 CPU 数量
vRAM	<ul style="list-style-type: none">• 呈现给来宾操作系统 (OS) 的内存量• 可以根据需要更改内存大小
虚拟磁盘	<ul style="list-style-type: none">• 存储虚拟机的操作系统和应用程序数据• 虚拟机应至少具有一个虚拟磁盘
vNIC	<ul style="list-style-type: none">• 使得虚拟机可以连接至其他物理机和虚拟机
虚拟 DVD/CD-ROM 驱动器	<ul style="list-style-type: none">• 它将虚拟机的 DVD/CD-ROM 驱动器映射至物理驱动器或 .iso 文件
虚拟软盘驱动器	<ul style="list-style-type: none">• 它将虚拟机的软盘驱动器映射至物理驱动器或 .flp 文件
虚拟 SCSI 控制器	<ul style="list-style-type: none">• 虚拟机使用虚拟 SCSI 控制器来访问虚拟磁盘
虚拟 USB 控制器	<ul style="list-style-type: none">• 将虚拟机的 USB 控制器映射至物理 USB 控制器



虚拟机控制台

- 提供鼠标、键盘和屏幕功能
- 向虚拟机 (VM) 发送电源变化（开/关）
- 允许访问虚拟机的 BIOS
- 通常用于虚拟硬件配置和故障排除事宜

虚拟化的资源管理

资源管理

从单个物理机或群集物理机向虚拟机 (VM) 分配资源以优化资源利用率的过程。

- 资源管理的目标
 - ▶ 控制资源利用率
 - ▶ 防止虚拟机独占资源
 - ▶ 基于虚拟机的相对优先级分配资源
- 必须池化资源以集中进行管理

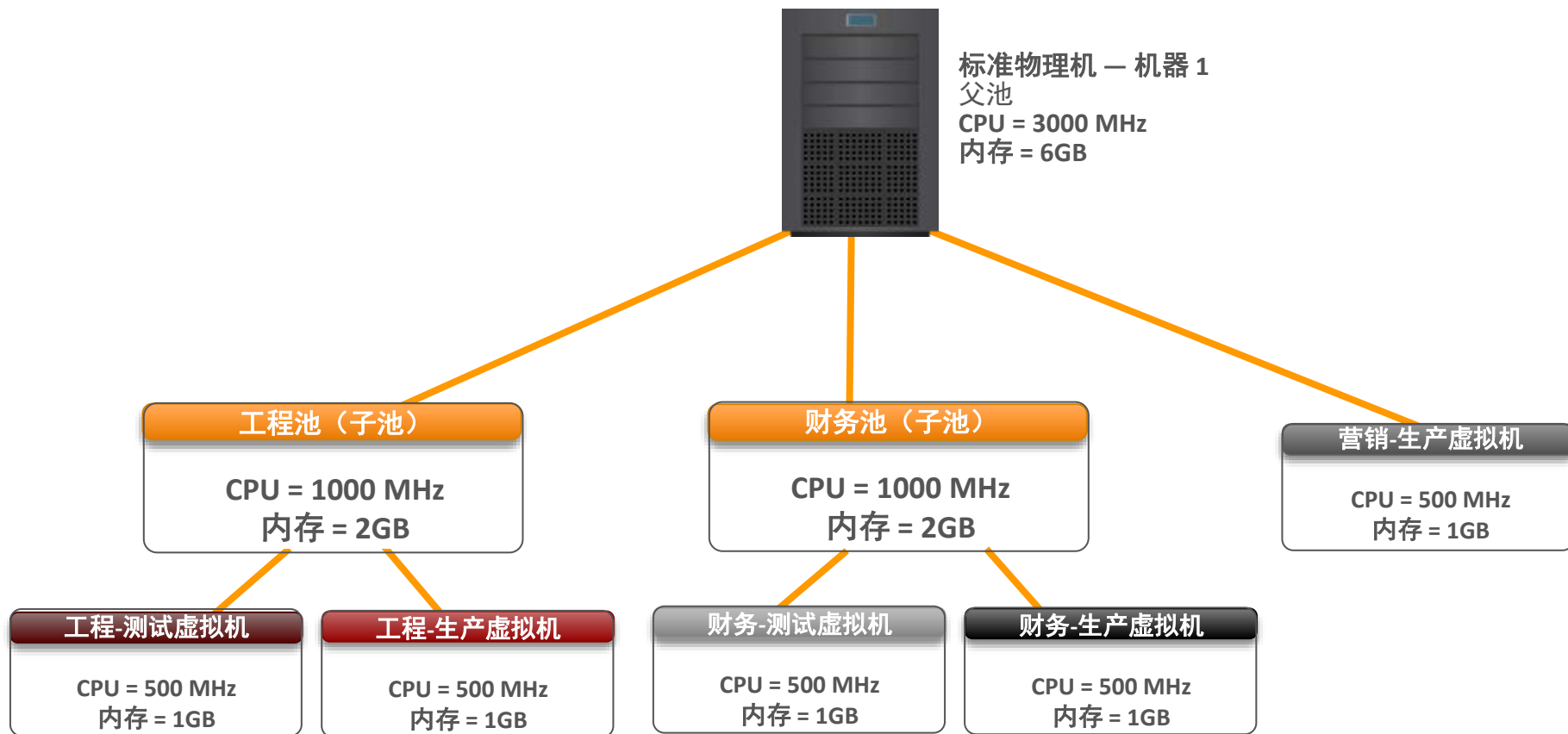
资源池

资源池

它被集中管理的聚合物理资源的逻辑抽象。

- 从物理机或群集创建而来
- 管理员可以从父资源池创建子资源池或虚拟机 (VM)
- 使用保留、限制和共享来控制资源池或虚拟机所消耗的资源

资源池示例



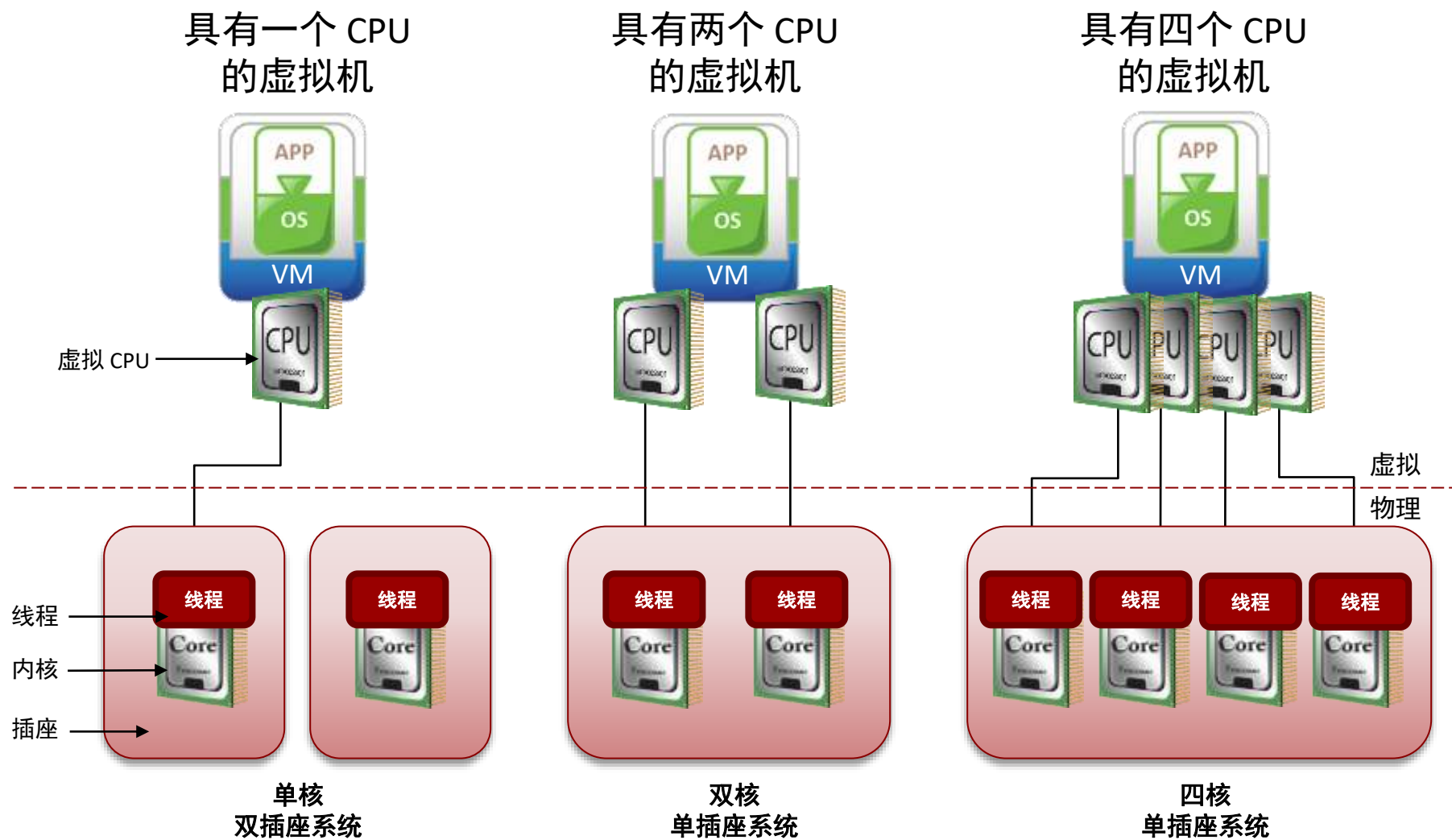
共享、限制和保留

- 用来控制子资源池或虚拟机 (VM) 所消耗的资源参数包括：
 - ▶ 共享
 - ▶▶ 相对其父池内的总资源量，VM 或子资源池可以拥有的 CPU 或内存资源量
 - ▶▶ 子资源池：当资源处于稀缺状态和发生资源争夺时，共享值定义父池中的子资源池的相对优先级。
 - ▶▶ 虚拟机：与子资源池类似，共享指定虚拟机的相对优先级。如果某个虚拟机具有两倍于另一个虚拟机的 CPU/内存共享，则在虚拟机竞争资源时，它将有权消耗两倍的 CPU/内存。
 - ▶ 限制
 - ▶▶ 虚拟机或子资源池可以消耗的最大 CPU 和内存量
 - ▶▶ 子资源池：它定义子资源池被允许消耗的最大 CPU (MHz) 和内存量 (MB)。
 - ▶▶ 虚拟机：与资源池类似，限制定义虚拟机被允许消耗的最大 CPU (MHz) 和内存量 (MB)。虚拟机可以消耗的最大内存和 CPU 量是在创建虚拟机时配置的，并只能在虚拟机关机后修改。
 - ▶ 保留
 - ▶▶ 为虚拟机或子资源池保留的 CPU 和内存量
 - ▶▶ 子资源池：它定义为子资源池保留的 CPU (MHz) 和内存量 (MB)。
 - ▶▶ 虚拟机：与子资源池类似，保留定义为虚拟机保留的 CPU (MHz) 和内存量 (MB)。如果虚拟机没有用到它所保留的 CPU 和内存总量，未使用的部分可供其他虚拟机使用，直至该虚拟机需要时。如果池内没有提供保留所定义的资源量，虚拟机将不会开机。

优化 CPU 资源

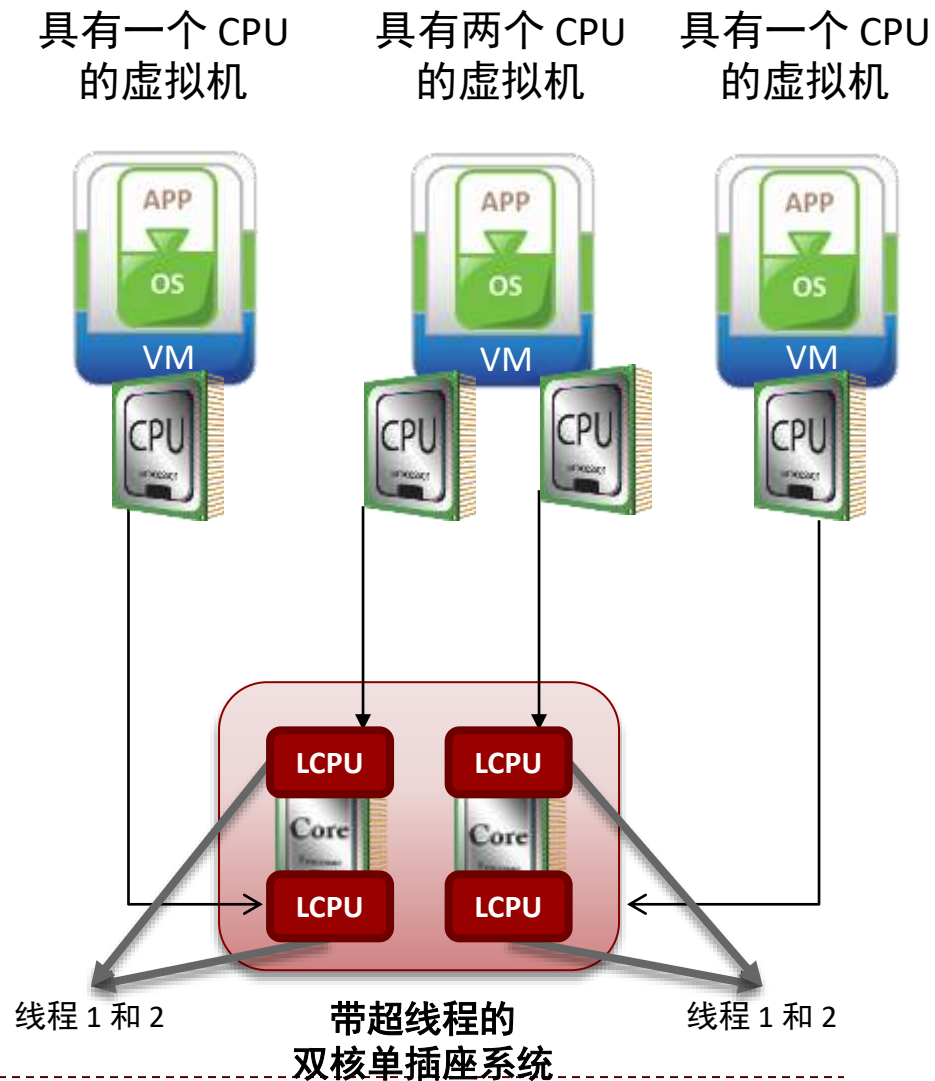
- 现代 CPU 配备多个内核和超线程
 - ▶ 多核处理器在单个 CPU 内具有多个处理单元（内核）
 - ▶ 超线程使得物理 CPU 显示为两个或更多个逻辑 CPU
- 高效和公平地分配 CPU 资源至关重要
- 虚拟机管理程序调度物理 CPU 上的虚拟 CPU
- 虚拟机管理程序支持多核、超线程和 CPU 负载均衡功能，以优化 CPU 资源

多核处理器



超线程

- 使得物理 CPU 显示为两个逻辑 CPU (LCPU)
 - ▶ 使得操作系统 (OS) 可以同时调度两个或更多个线程
- 两个 LCPU 共享相同的物理资源
 - ▶ 在当前线程暂停时, CPU 可以执行其他线程
- 在启用超线程的 CPU 上运行的虚拟机管理程序可提供改进的性能和利用率



CPU 负载均衡

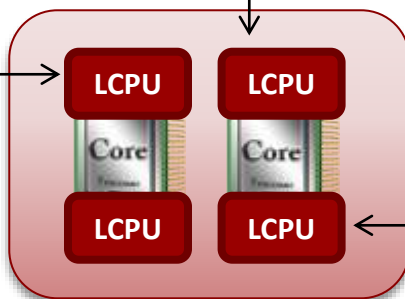
具有一个 CPU
的虚拟机



具有一个 CPU
的虚拟机



具有一个 CPU
的虚拟机



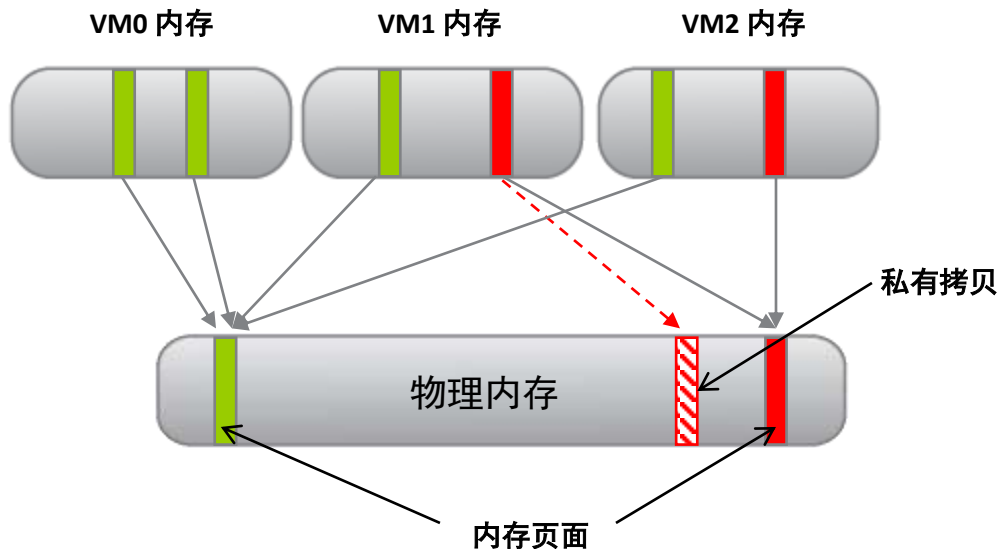
超线程双核
双插座系统

优化内存资源

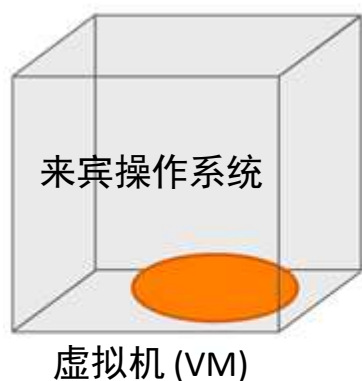
- 虚拟机管理程序管理机器的物理内存
 - ▶ 部分内存由虚拟机管理程序使用
 - ▶ 剩余部分供虚拟机 (VM) 使用
- 虚拟机可以配置比实际可用量更多的内存（称为“内存过量配置”）
 - ▶ 执行内存优化以允许过量配置
- 内存管理技术包括透明化页面共享、内存膨胀和内存交换

透明化页面共享

- 虚拟机管理程序检测到完全相同的虚拟机 (VM) 内存页面，并将其映射到同一物理页面
 - ▶ 共享时的状态为只读
- 对于写入操作，虚拟机管理程序将共享的页面处理为写入时拷贝
- 尝试在共享页面上写入
 - ▶ 生成次要页面故障
 - ▶ 写入后创建私有拷贝并重新映射内存

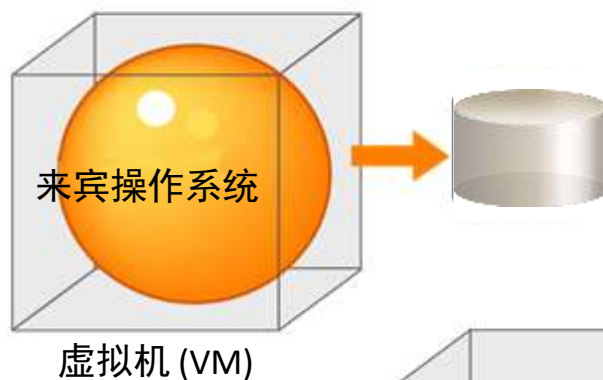


内存膨胀:



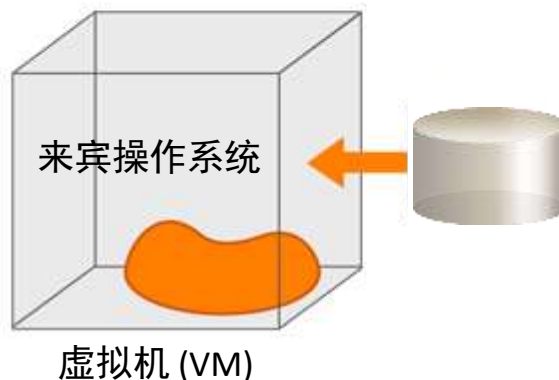
无内存短缺, 气球未膨胀

膨胀技术使得来賓操作系统可以释放部分虚拟机内存



1. 内存短缺, 气球膨胀
2. 驱动程序要求来賓操作系统 (OS) 提供内存
3. 来賓操作系统强制调出页面
4. 虚拟机管理程序回收内存

1. 内存短缺已解决, 气球萎缩
2. 驱动程序释放内存
3. 来賓操作系统可以使用页面
4. 虚拟机管理程序确认内存



内存交换

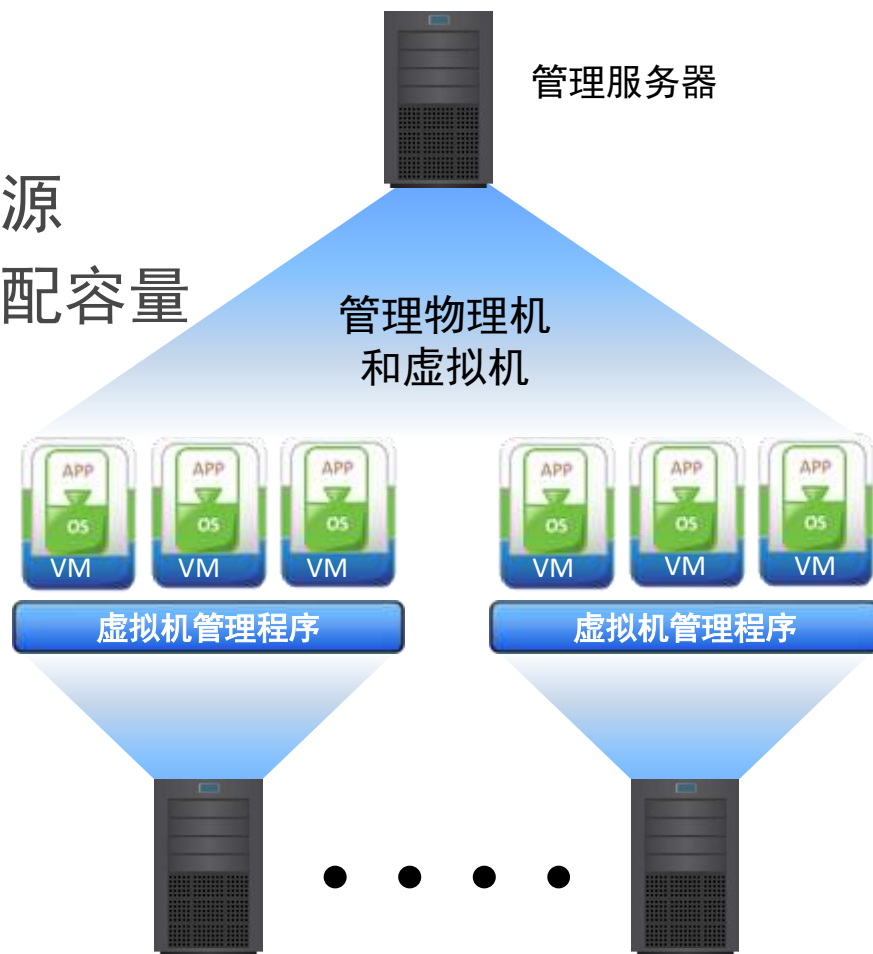
- 每个已开机的虚拟机 (VM) 都需要自己的交换文件
 - ▶ 在虚拟机开机时创建
 - ▶ 在虚拟机关机时删除
- 交换文件的大小等于内存限制与虚拟机内存保留之间的差值
- 内存稀缺时，虚拟机管理程序将换出虚拟机的内存内容
- 交换是最后选择，因为它将导致明显的性能影响

虚拟机亲和性

- 虚拟机对虚拟机亲和性：
 - ▶ 选定的虚拟机应该在同一虚拟机管理程序上运行
 - ▶▶ 在虚拟机之间存在大量通信时提高性能
 - ▶ 反亲和性确保选定的虚拟机不在同一虚拟机管理程序上（例如：由于可用性的原因）
- 虚拟机对物理服务器亲和性：
 - ▶ 指定选定的虚拟机是否只能放置在特定的虚拟机管理程序上（例如：由于许可的原因）
 - ▶ 反亲和性允许虚拟机在群集内的不同虚拟机管理程序上移动（例如：为满足高可靠性或性能需求）

资源管理工具

- 提供管理正在运行虚拟机管理程序的物理机的能力
- 实现从管理服务器集中管理资源
- 实现资源池化以及向虚拟机分配容量
 - ▶ 与虚拟机管理程序通信以执行管理
- 提供操作自动化

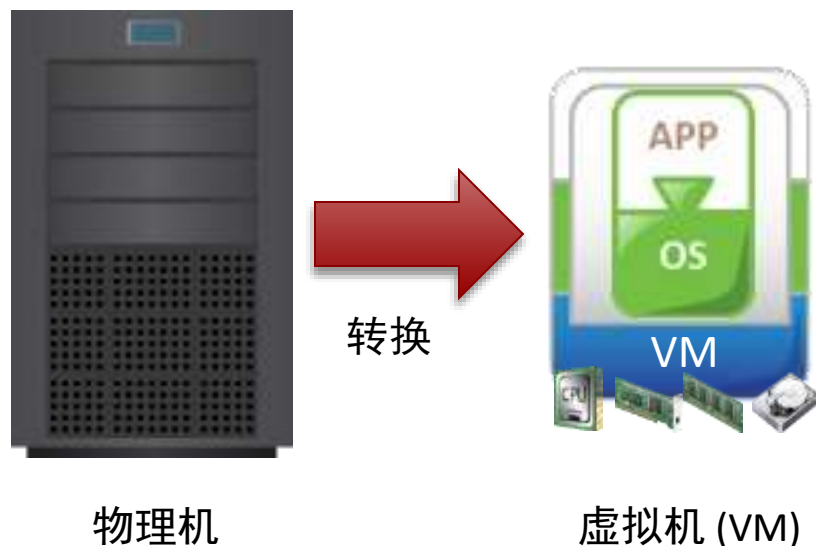


物理机到虚拟机 (P2V) 转换

P2V 转换

它是将物理机转换为虚拟机的 (VM) 的过程。

- 将数据从物理机磁盘克隆到虚拟机磁盘
- 执行目标虚拟机的系统重配置, 例如:
 - ▶ 更改 IP 地址和计算机名称
 - ▶ 安装必要的设备驱动程序以引导虚拟机



P2V 转换器的优势

- 降低设置新虚拟机 (VM) 所需的时间
- 无需重新安装操作系统 (OS) 或应用程序即可实现传统机器向新硬件的迁移
- 跨异质硬件执行迁移

转换选项

- 热转换

- ▶ 在物理机正在运行时发生
- ▶ 执行同步
 - ▶▶ 拷贝在初始克隆期间发生更改的数据块
- ▶ 在源上执行关机和在目标虚拟机 (VM) 上执行开机
- ▶ 更改选定机器的 IP 地址和机器名（如果两台机器必须在同一网络内共存）

- 冷转换

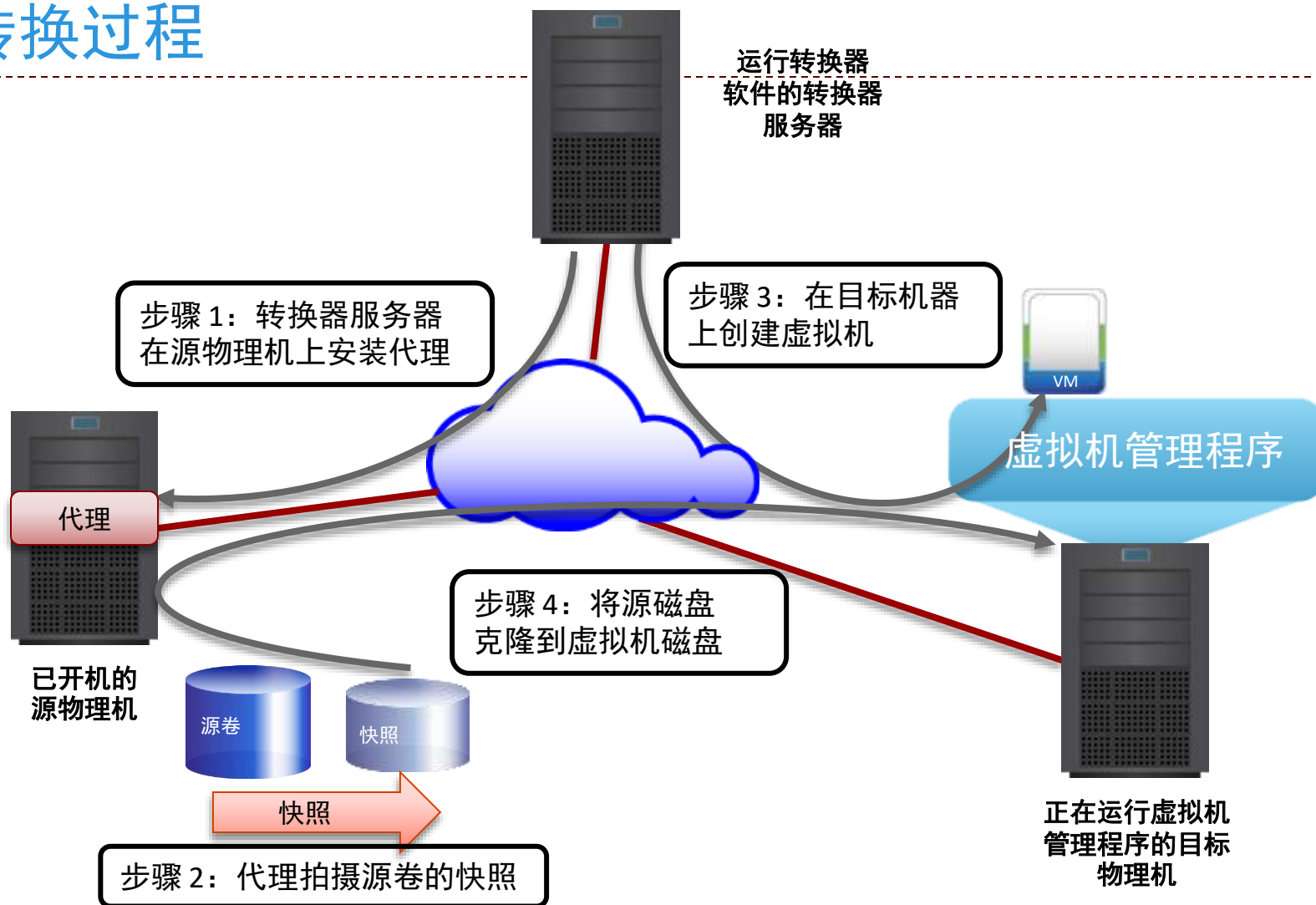
- ▶ 在物理机没有运行操作系统和应用程序时发生
- ▶ 使用转换器引导光盘引导物理机
- ▶ 创建一致的物理机拷贝



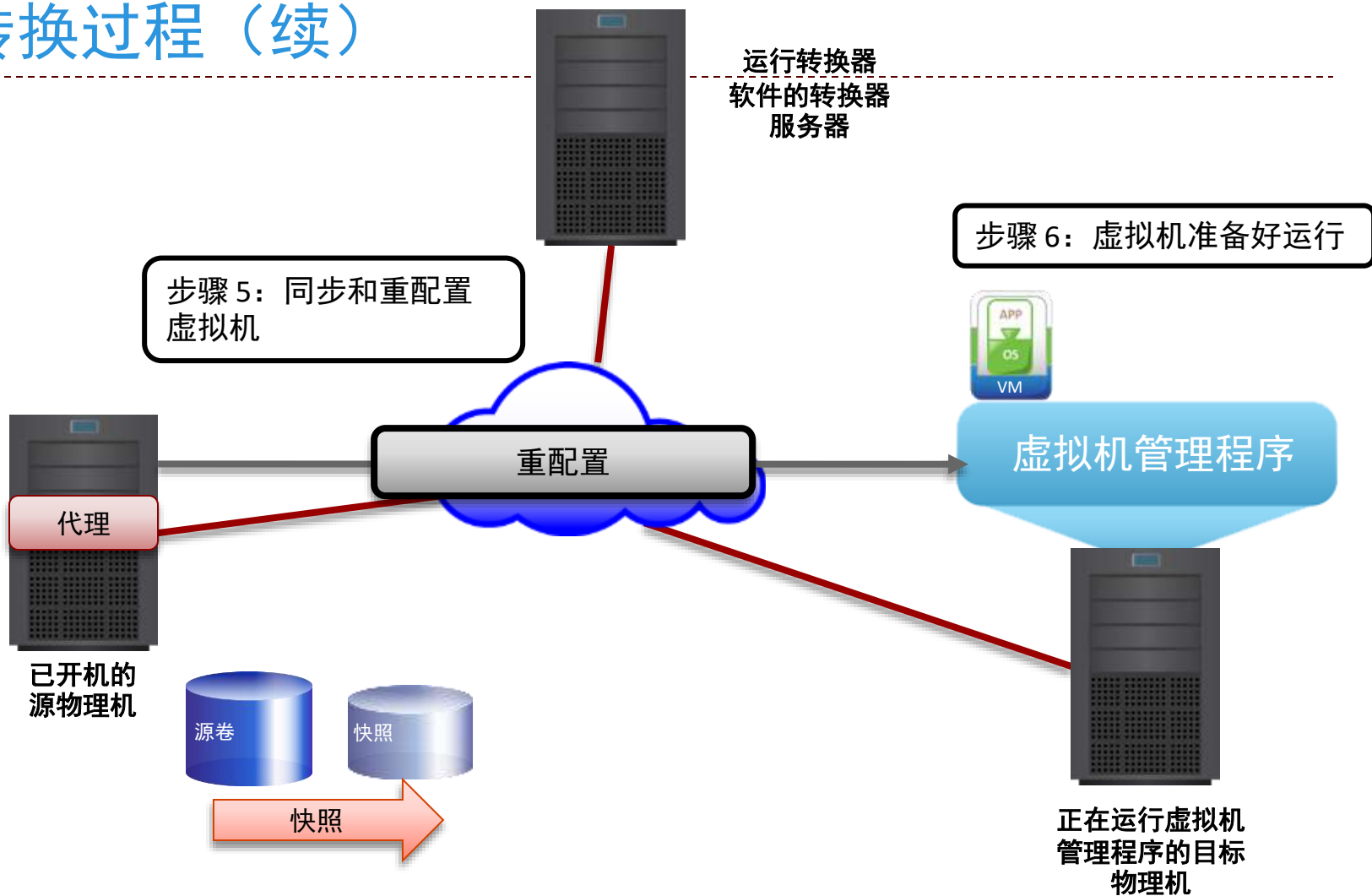
P2V 转换器的组件

- 有三个关键组件：
 - ▶ 转换器服务器
 - ▶▶ 负责控制转换过程
 - ▶▶ 仅用于热转换（在源正在运行自己的操作系统时）
 - ▶▶ 在源机器上推送和安装代理
 - ▶ 转换器代理
 - ▶▶ 负责执行转换
 - ▶▶ 仅用于热模式
 - ▶▶ 安装在物理机上，以将其转换为虚拟机 (VM)
 - ▶ 转换器引导光盘
 - ▶▶ 包含其操作系统和转换器应用程序的可引导光盘
 - ▶▶ 转换器应用程序用于执行冷转换

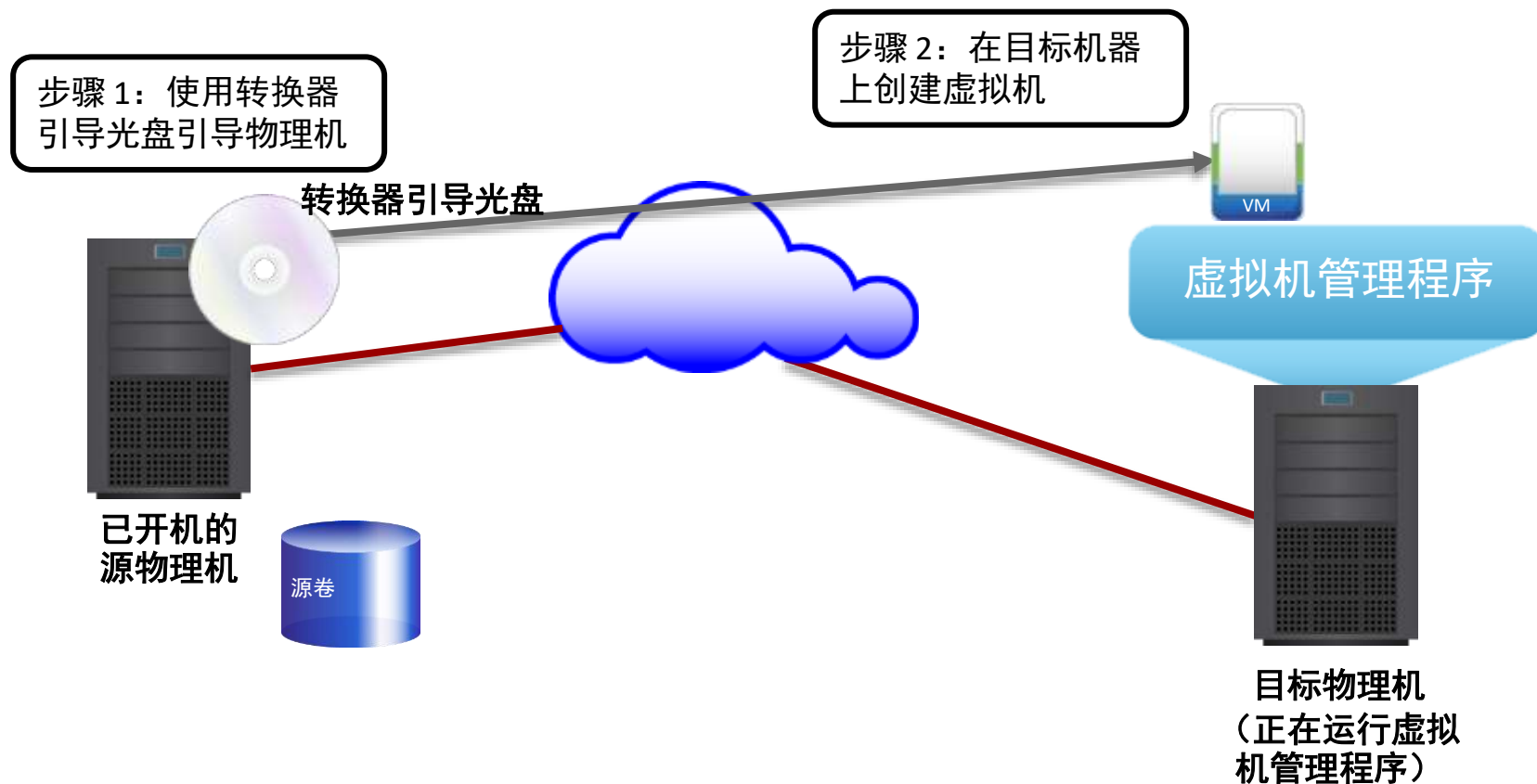
热转换过程



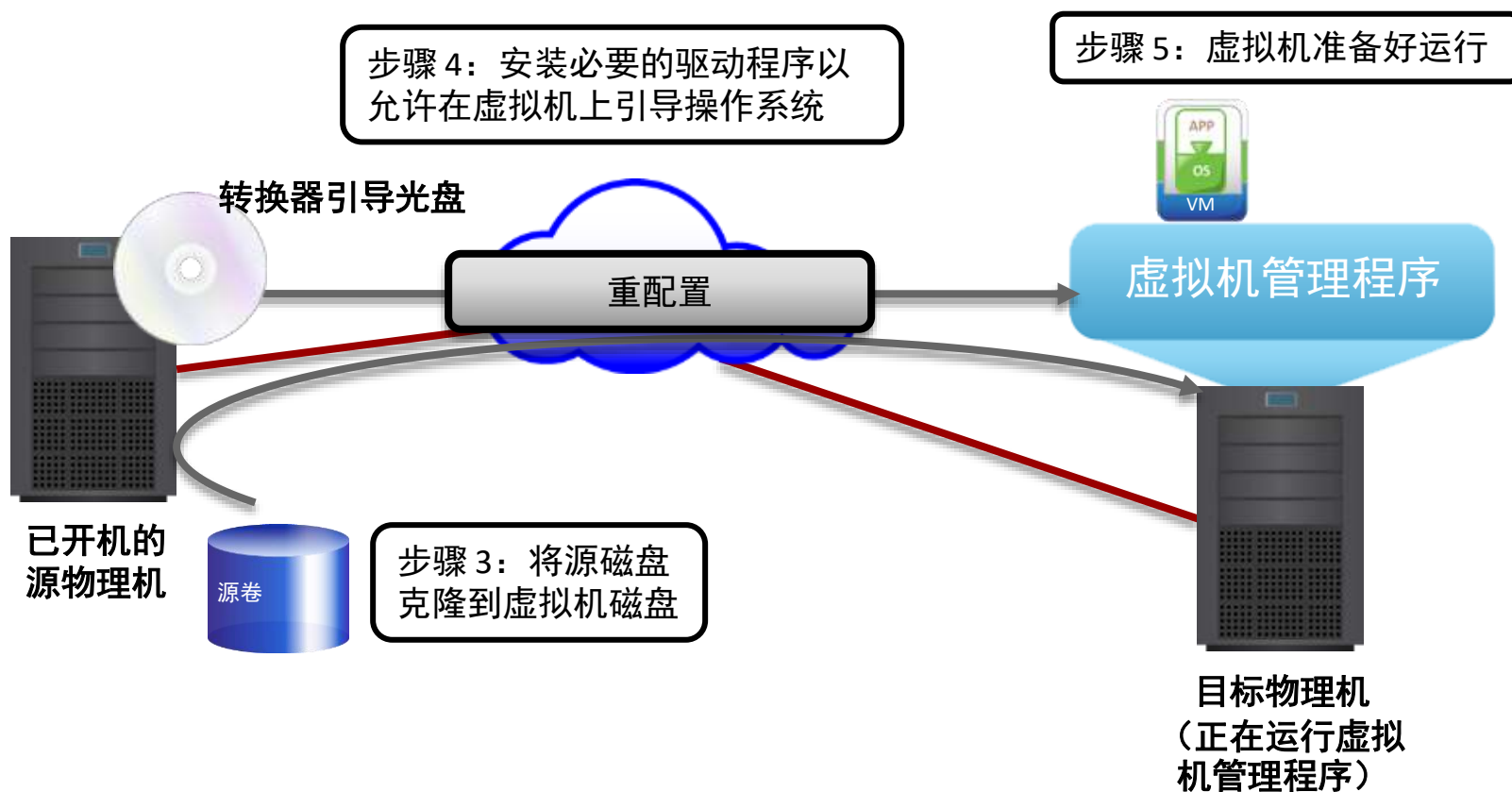
热转换过程（续）



冷转换过程



冷转换过程（续）



存储虚拟化

存储

存储

指持久存储数据以供后续使用的资源。

- 个人/企业创建的数据必须存储以便进一步处理
- 所用存储设备的类型取决于数据类型以及数据的创建速度和使用频率
- 存储设备可以使用磁、光或固态介质
 - ▶ 示例：磁盘驱动器（磁）、CD（光）、闪存驱动器（固态）

存储设备选件

磁带机

用于长期数据存储的低成本解决方案
顺序数据访问、物理磨损以及存储/检索开销

光盘

一写多读 (WORM): CD、DVD
容量和速度均有限
广泛应用于小型单用户环境中

磁盘驱动器

随机读/写访问
使用机械部件进行数据访问
应用最广泛的大存储容量存储设备

固态驱动器

提供任务关键型应用程序所需的超高性能
每个 I/O 延迟极低, 电源要求低且每个驱动器吞吐量极高

独立磁盘冗余阵列 (RAID)

RAID

一项将多个磁盘驱动器用作一个集合，以提供保护、容量和/或性能优势的技术

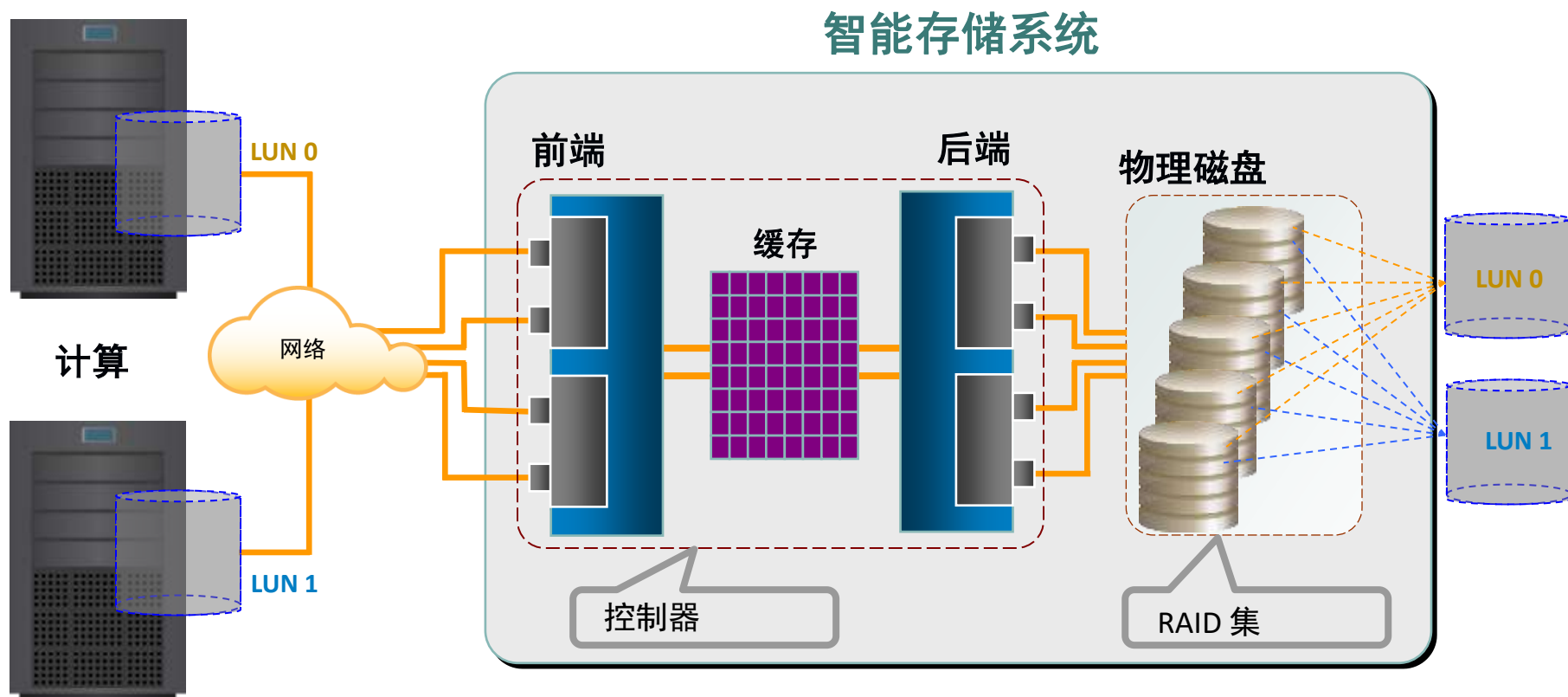
- 突破磁盘驱动器的限制
- 提高存储系统性能
 - ▶ 同时从多个磁盘提供 I/O 服务
- RAID 技术：
 - ▶ 分条、镜像和奇偶校验

RAID 级别

RAID 级别	定义
RAID 0	分条，无容错
RAID 1	磁盘镜像
嵌套	RAID 级别组合，如：RAID 1 + RAID 0
RAID 3	奇偶校验 RAID，使用专用奇偶校验磁盘
RAID 5	奇偶校验 RAID，奇偶校验分布在集合中的所有磁盘上
RAID 6	具有双奇偶校验的分布式奇偶校验 RAID



存储和计算系统的结合



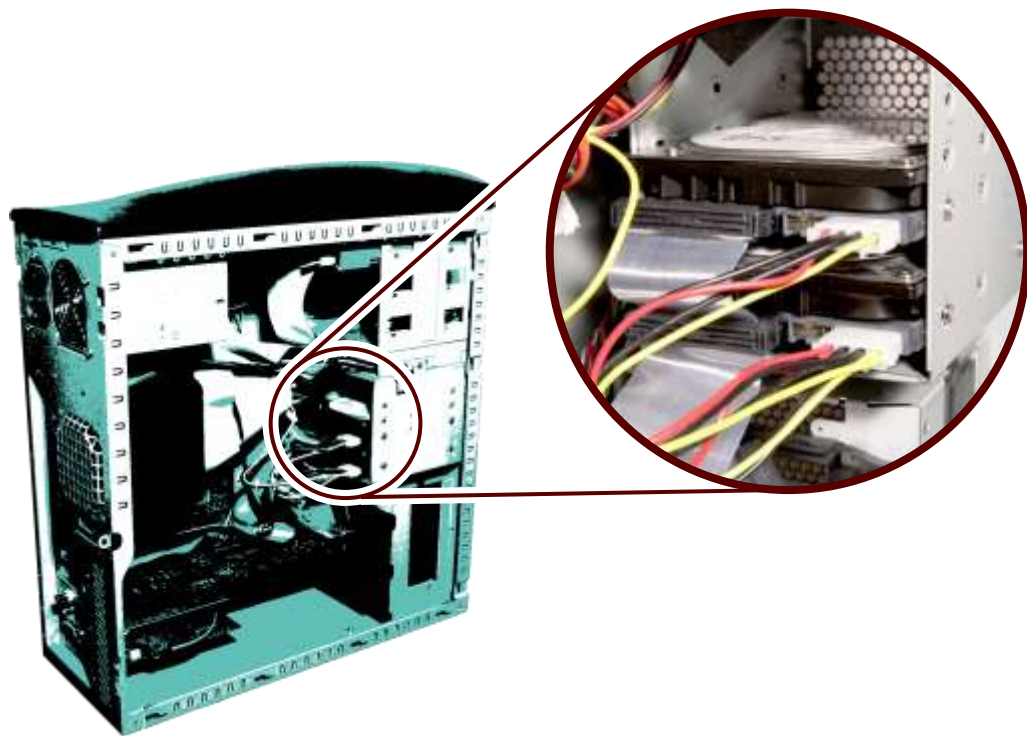
直接连接存储 (DAS)

DAS

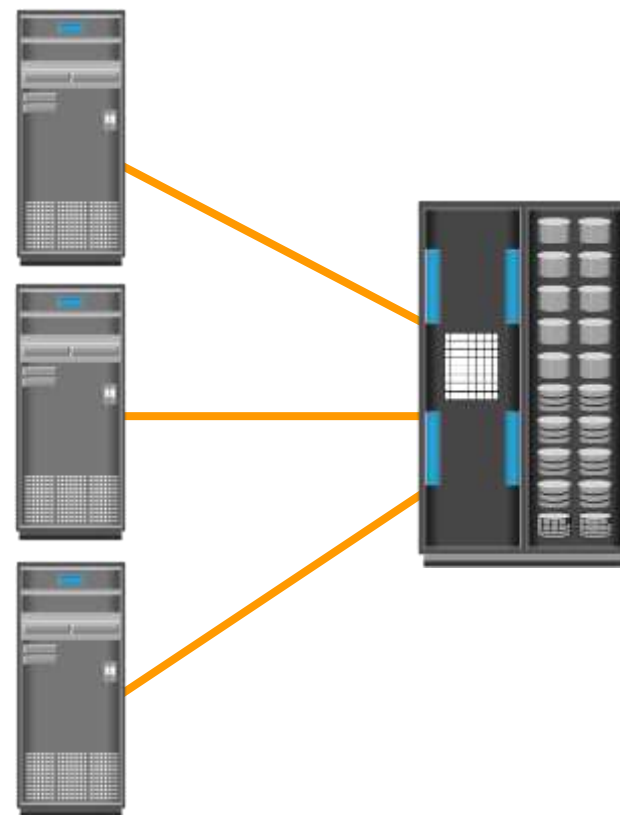
与计算系统直接连接的内部或外部存储设备

- 根据与计算系统相关的存储设备的位置，DAS 分为内部 DAS 或外部 DAS
- 优势：
 - ▶ 易于部署，适用于本地数据调配
 - ▶ 资本支出低，复杂性低
- 挑战：
 - ▶ 可扩展性有限
 - ▶ 共享资源能力有限
 - ▶▶ 存在过度利用和未充分利用存储资源岛

直连存储 (DAS)



内部直接连接



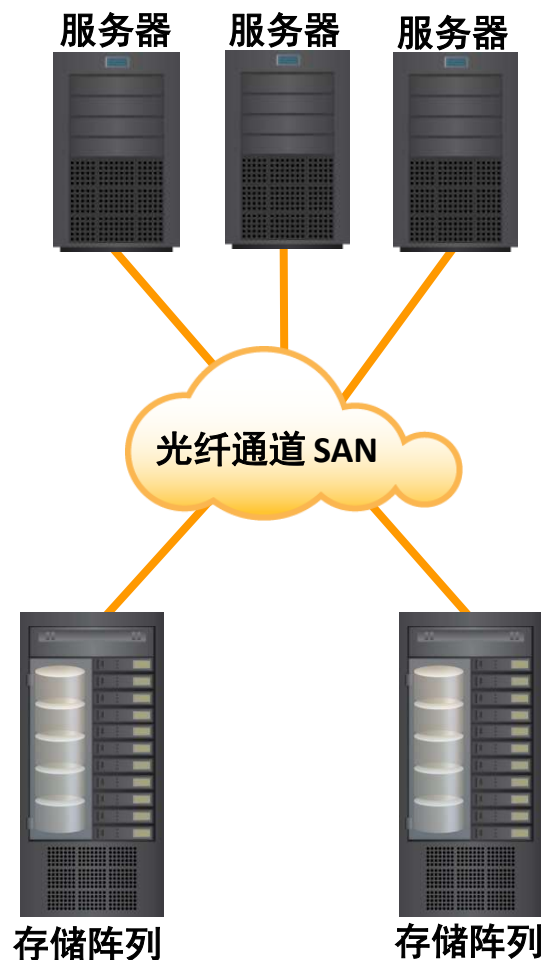
外部直接连接

光纤SAN (Storage Area Network)

- 计算系统和共享存储设备的专用高速网络
- 采用基于光纤通道的 SCSI 协议
- 提供数据块级数据访问

优势

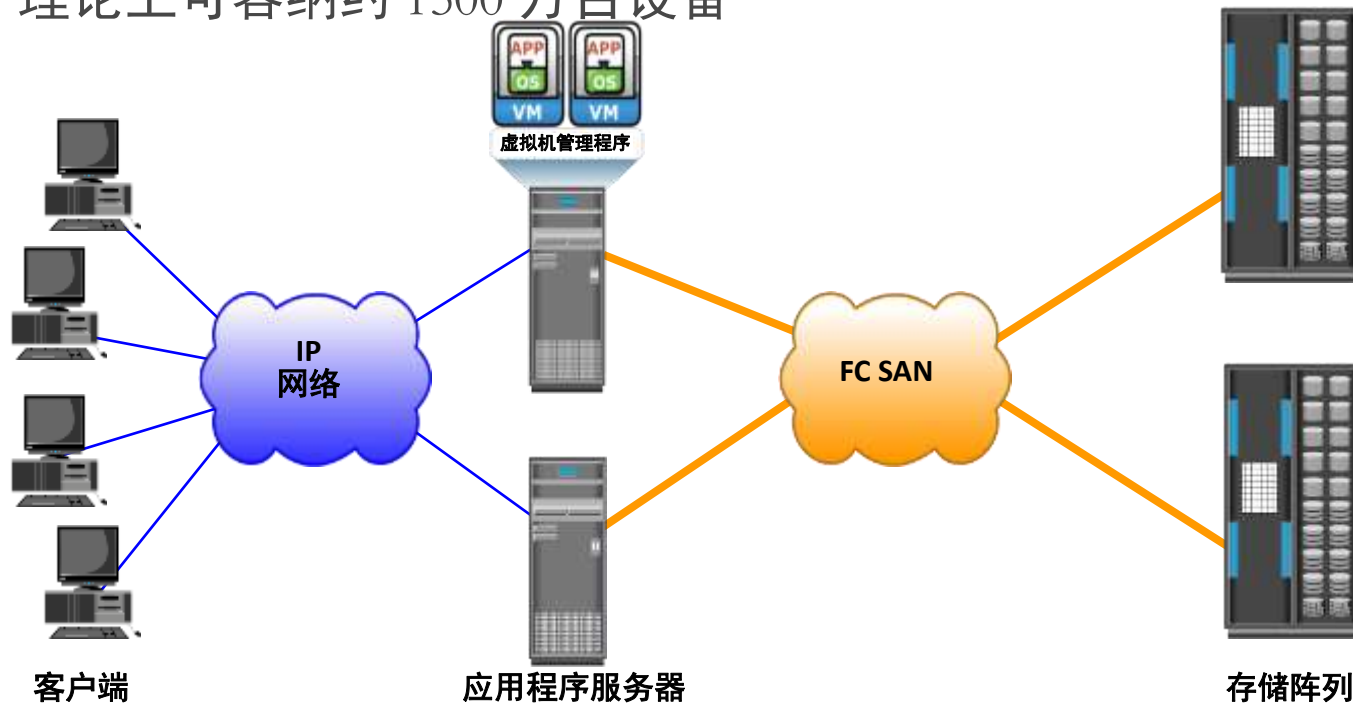
- 可实现存储整合和共享
- 可实现集中化管理
- 提供可扩展性和出色性能
- 减少存储和管理成本



了解光纤通道

注意：FibRE 指协议，而 fibER 指介质。

- 光纤通道是在高速光缆和串行铜缆上运行的高速网络技术。
 - 最新 FC 实施最高支持 16 Gb/s 的速度
- 高可扩展性
 - ▶ 理论上可容纳约 1500 万台设备

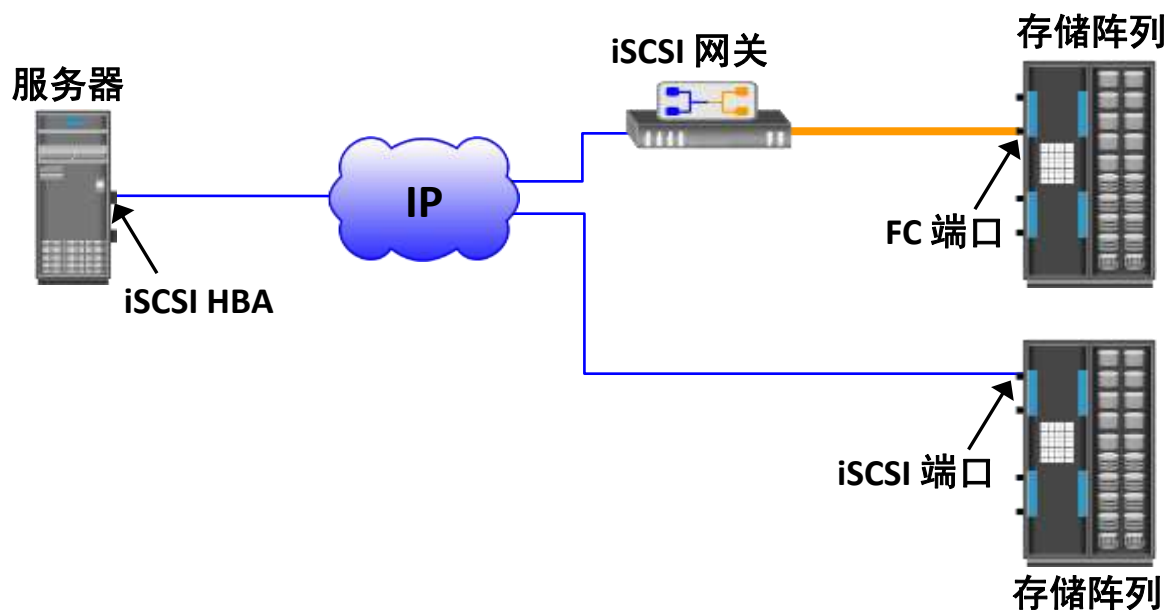


IP SAN 协议：iSCSI

- 用于连接主机和存储的基于 IP 的协议
- 将 SCSI 命令和数据封装到 IP 数据包并使用 TCP/IP 传输它们

iSCSI 的组件

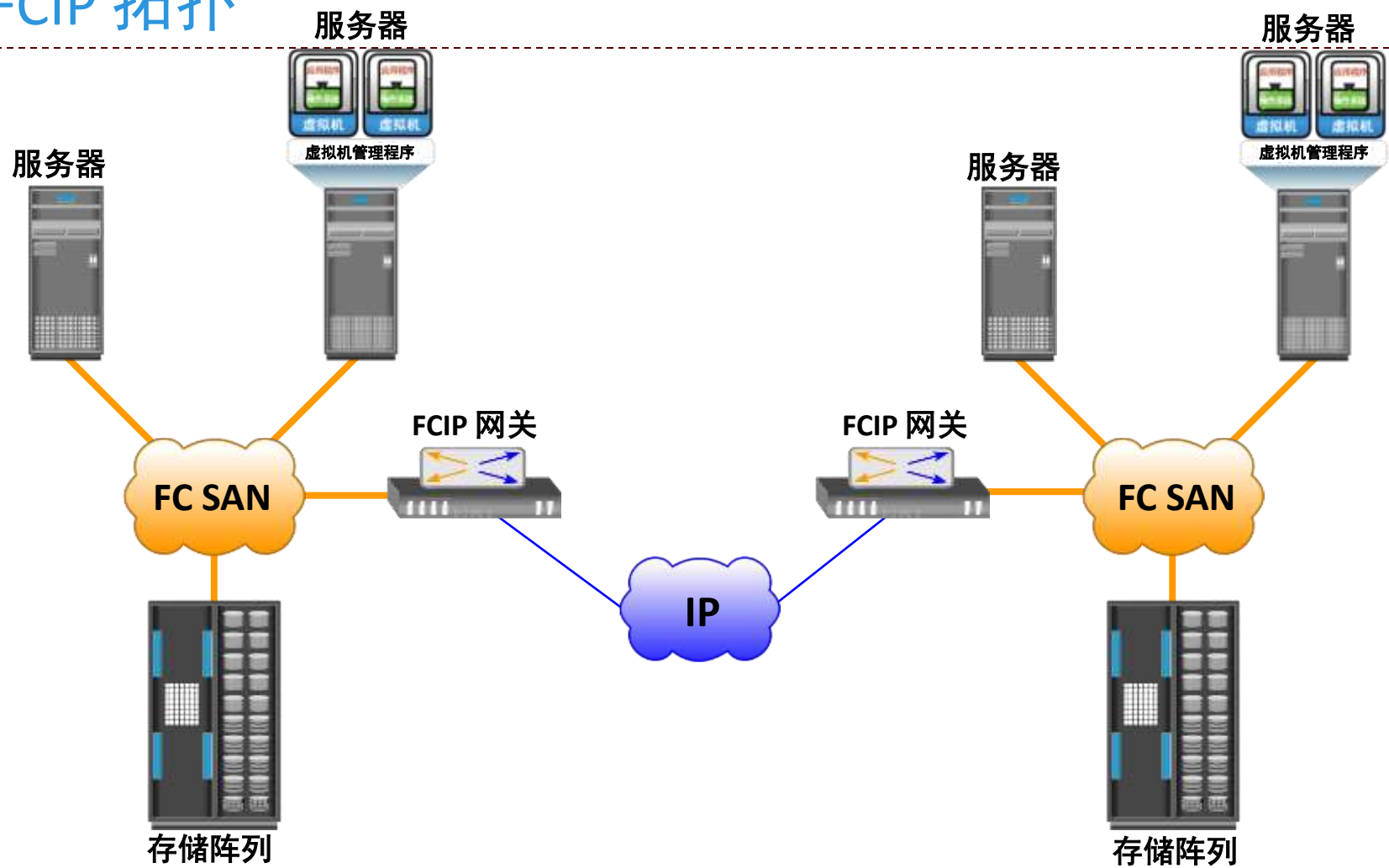
- iSCSI 启动器
 - ▶ 示例：iSCSI HBA
- iSCSI 目标
 - ▶ 具有 iSCSI 端口的存储阵列
 - ▶ iSCSI 网关 – 支持与 FC 存储阵列的通信
- IP 网络



IP SAN 协议：FCIP——远距离分布式FC的IP连接

- 用于连接分布式 FC SAN 岛的基于 IP 的协议：通过可靠的高速链路对地理上分散的 SAN 进行互连。
- 此方法涉及通过 IP 基础架构传输 FC 数据块数据。
 - 将 FC 帧封装到 IP 数据包
 - 通过现有 IP 网络（用于在不同 FC SAN 之间传输 FC 数据）创建虚拟 FC 链路
- 它兼具以下两方面的优势：FC SAN 和经验证且广泛部署的 IP 基础架构
 - FCIP 广泛用于灾难恢复实施，其中，数据将会复制到位于远程站点的存储中。

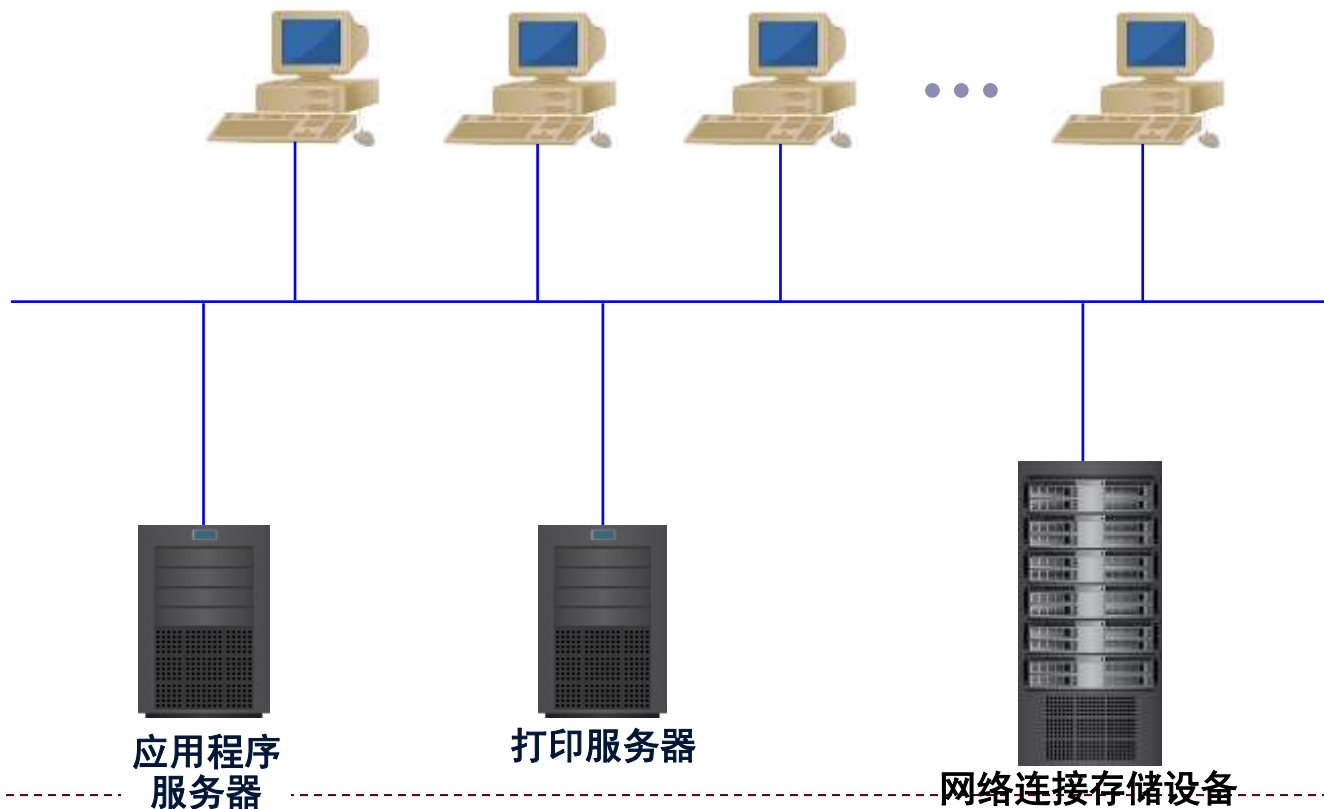
FCIP 拓扑



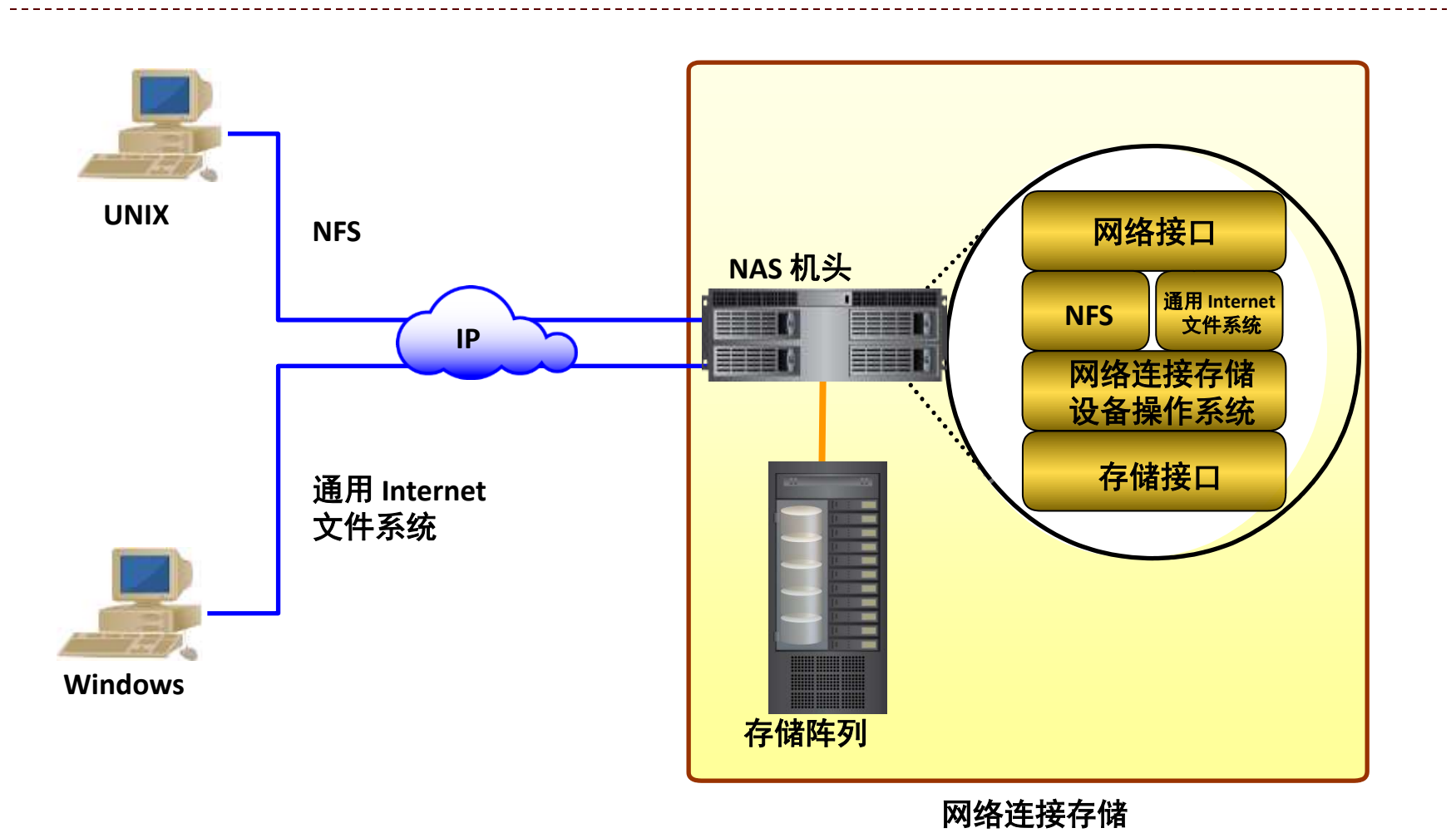
什么是网络连接存储 (NAS)?

网络连接存储

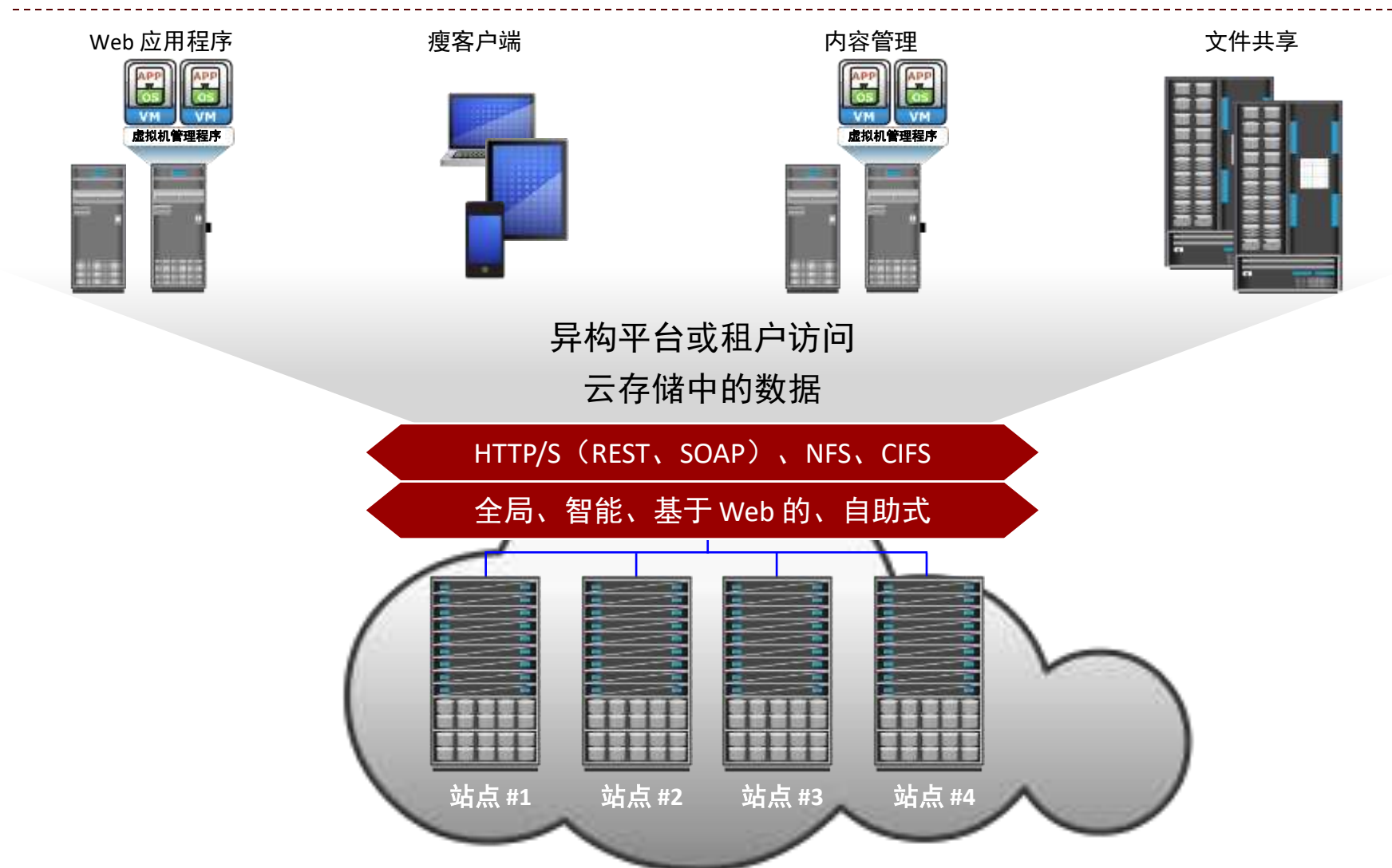
与网络连接以提供对异构客户端的文件级数据访问的存储设备



网络连接存储的组件



基于对象的存储



基于云的存储

存储虚拟化

存储虚拟化

它是掩蔽底层物理存储资源的复杂性并将这些资源的逻辑视图呈现给计算系统的过程。

- 虚拟化层执行逻辑到物理的存储映射
- 虚拟化层提取物理存储设备的标识
 - ▶ 从多个异构存储阵列创建存储池
- 从存储池创建虚拟卷，并将虚拟卷分配给计算系统

存储虚拟化的优势

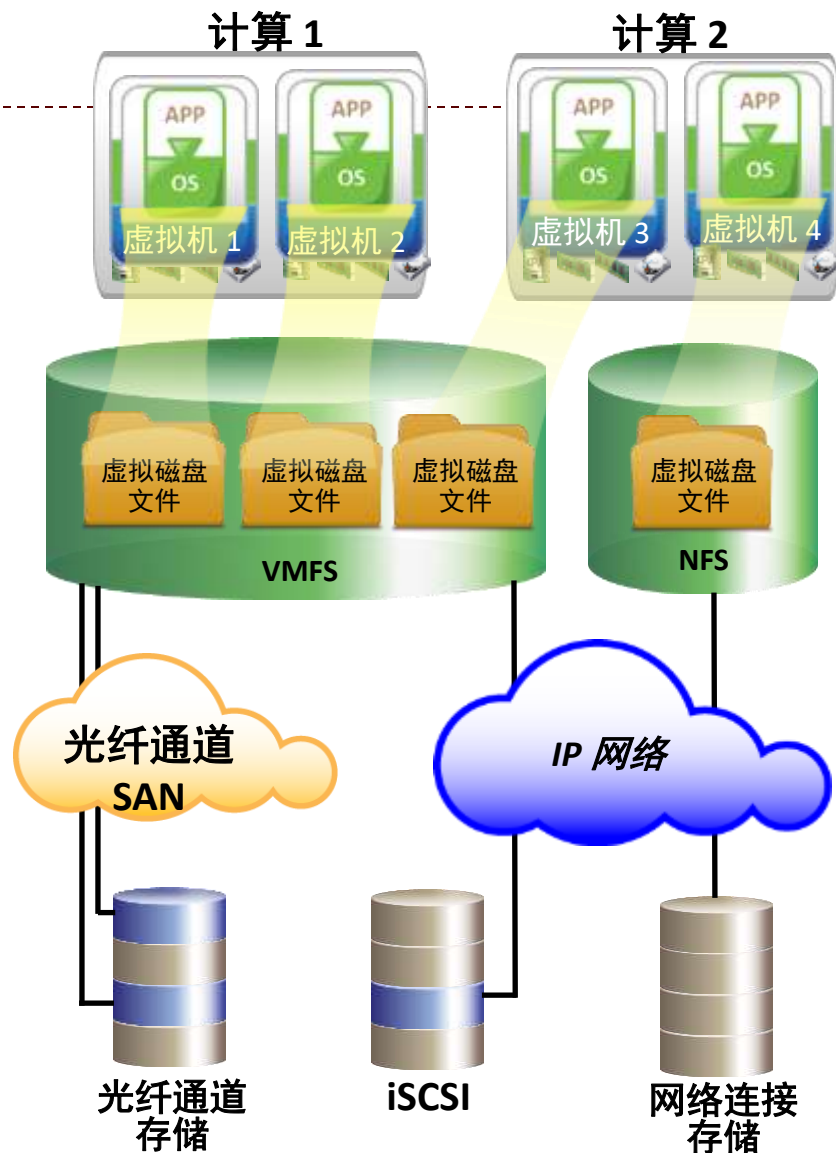
- 无需任何宕机即可添加或删除存储——弹性伸缩
- 提高存储利用率，因此降低总体拥有成本——重复利用
- 在存储设备之间提供无中断的数据迁移
- 支持异构的多供应商存储平台
- 简化存储管理——高可管理性

不同层的存储虚拟化

层	示例
计算	<ul style="list-style-type: none">• 虚拟机的存储资源调配
网络	<ul style="list-style-type: none">• 数据块级别的虚拟化• 文件级别的虚拟化
存储	<ul style="list-style-type: none">• 虚拟资源调配• 自动存储分层

虚拟机的存储

- 虚拟机作为一组文件存储在虚拟机管理程序可访问的存储空间上
- “虚拟磁盘文件”表示虚拟机用来存储其数据的虚拟磁盘
- 虚拟磁盘文件的大小表示为虚拟磁盘分配的存储空间
- 虚拟机一直察觉不到
 - ▶ 虚拟机管理程序可用的总空间
 - ▶ 底层存储技术

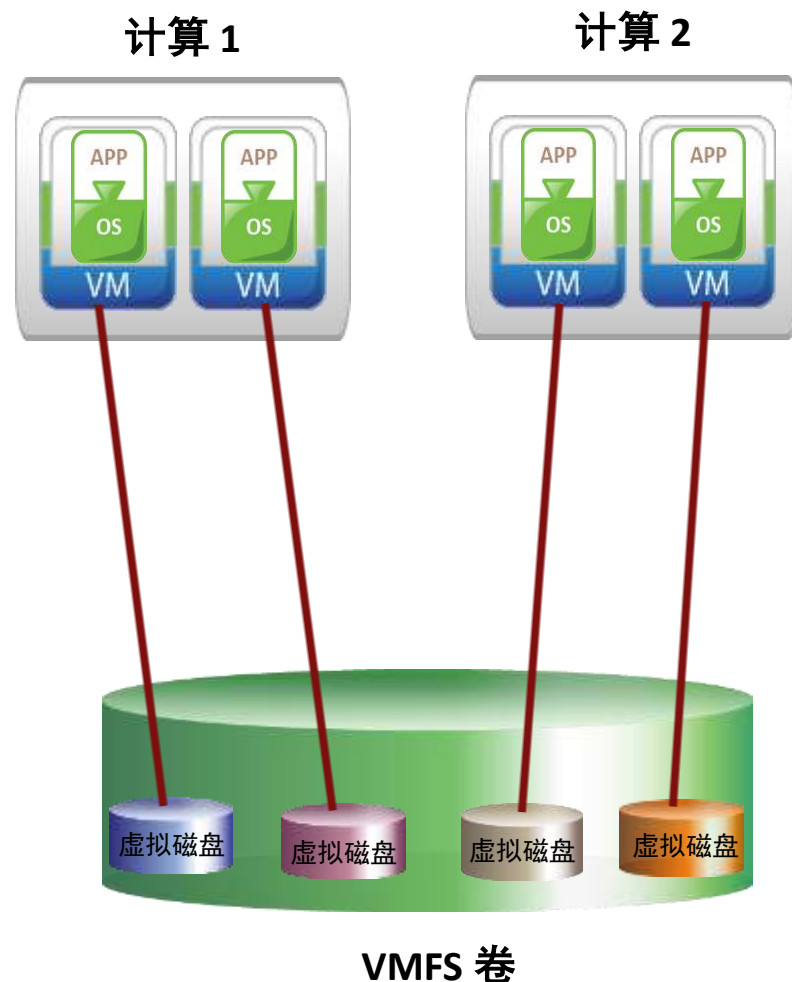


用于管理虚拟机文件的文件系统

- 虚拟机管理程序使用两个文件系统管理虚拟机文件
 - ▶ 称为虚拟机文件系统 (VMFS) 的虚拟机管理程序的本机文件系统
 - ▶ 诸如网络连接存储文件系统的网络文件系统 (NFS)

虚拟机文件系统 (VMFS)

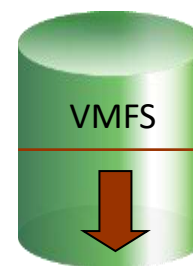
- 用于管理虚拟机文件的虚拟机管理程序本机文件系统
- 群集文件系统
 - ▶ 可以同时由多个计算系统访问
 - ▶ 提供磁盘上的锁定
- 使用 VMFS 卷存储虚拟机文件



VMFS 的动态扩展

- VMFS 可以在不中断虚拟机运行的情况下动态扩展

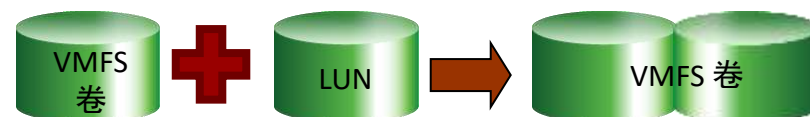
在现有卷上
扩展 VMFS



扩展 VMFS 的方法

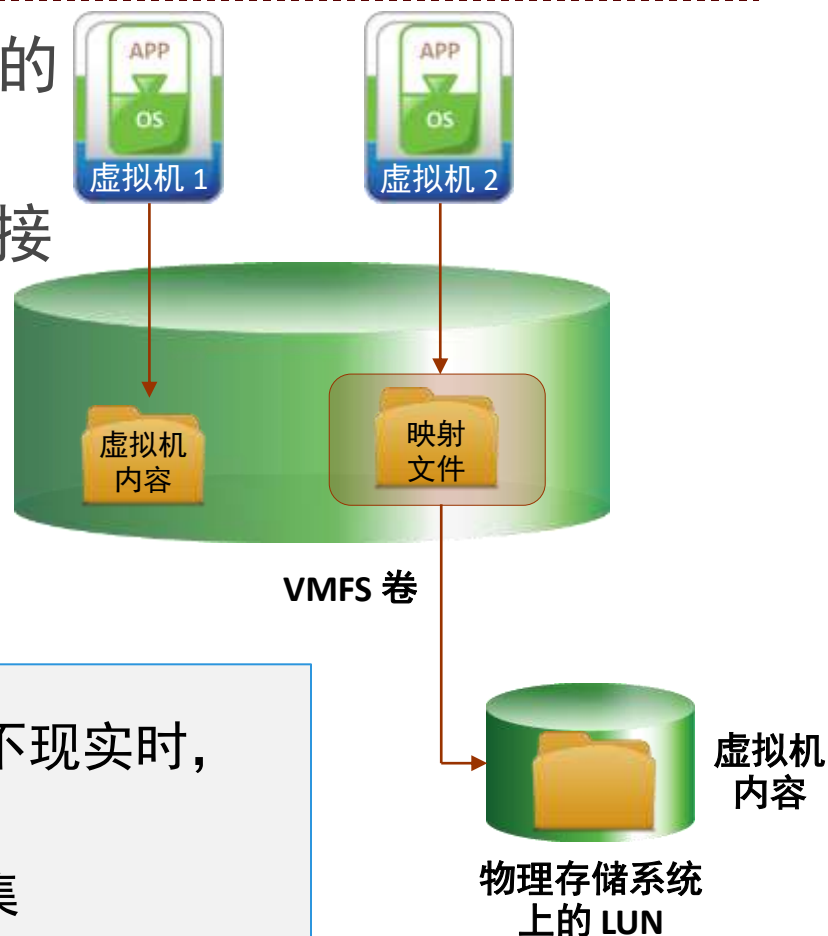
- 在 VMFS 所在的卷分区上动态扩展 VMFS
- 将一个或多个 LUN 添加到源 VMFS 卷

将 LUN 添加到现有 VMFS 卷



原始设备映射

- 允许虚拟机直接访问存储系统中的 LUN
- 在 VMFS 卷上包含 LUN 的符号链接
 - ▶ 充当允许直接访问 LUN 的代理

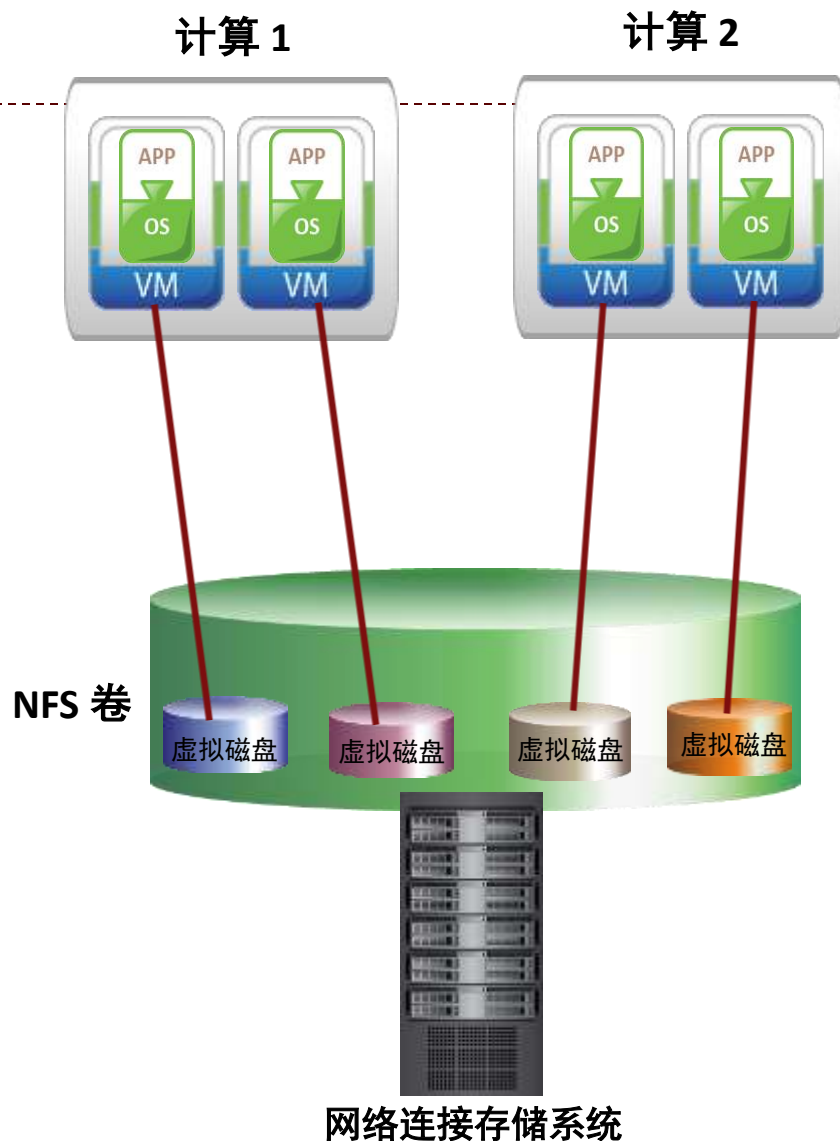


优势

- 将 LUN 上的大量数据移到虚拟磁盘不现实时，提供解决方案
- 允许使用物理计算机建立虚拟机群集

网络文件系统

- 虚拟机管理程序使用 NFS 协议访问网络连接存储文件系统
- 在网络连接存储设备上创建 NFS 卷
 - ▶ 向虚拟机提供存储
 - ▶ 同时由多个计算系统访问

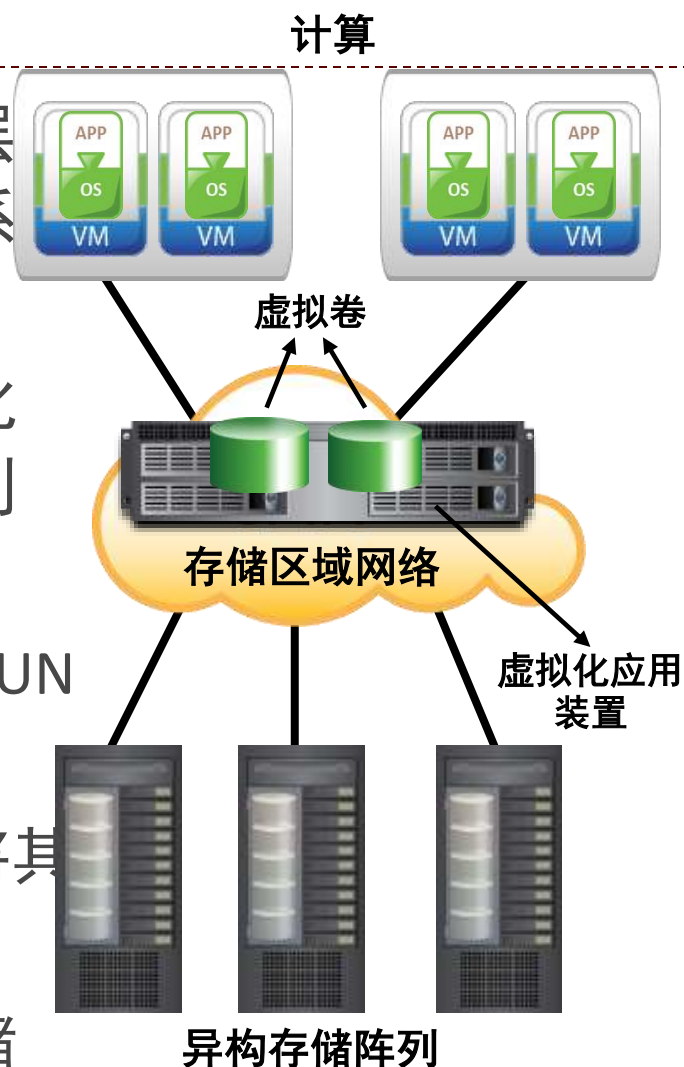


数据块级别和文件级别的虚拟化 — 概述

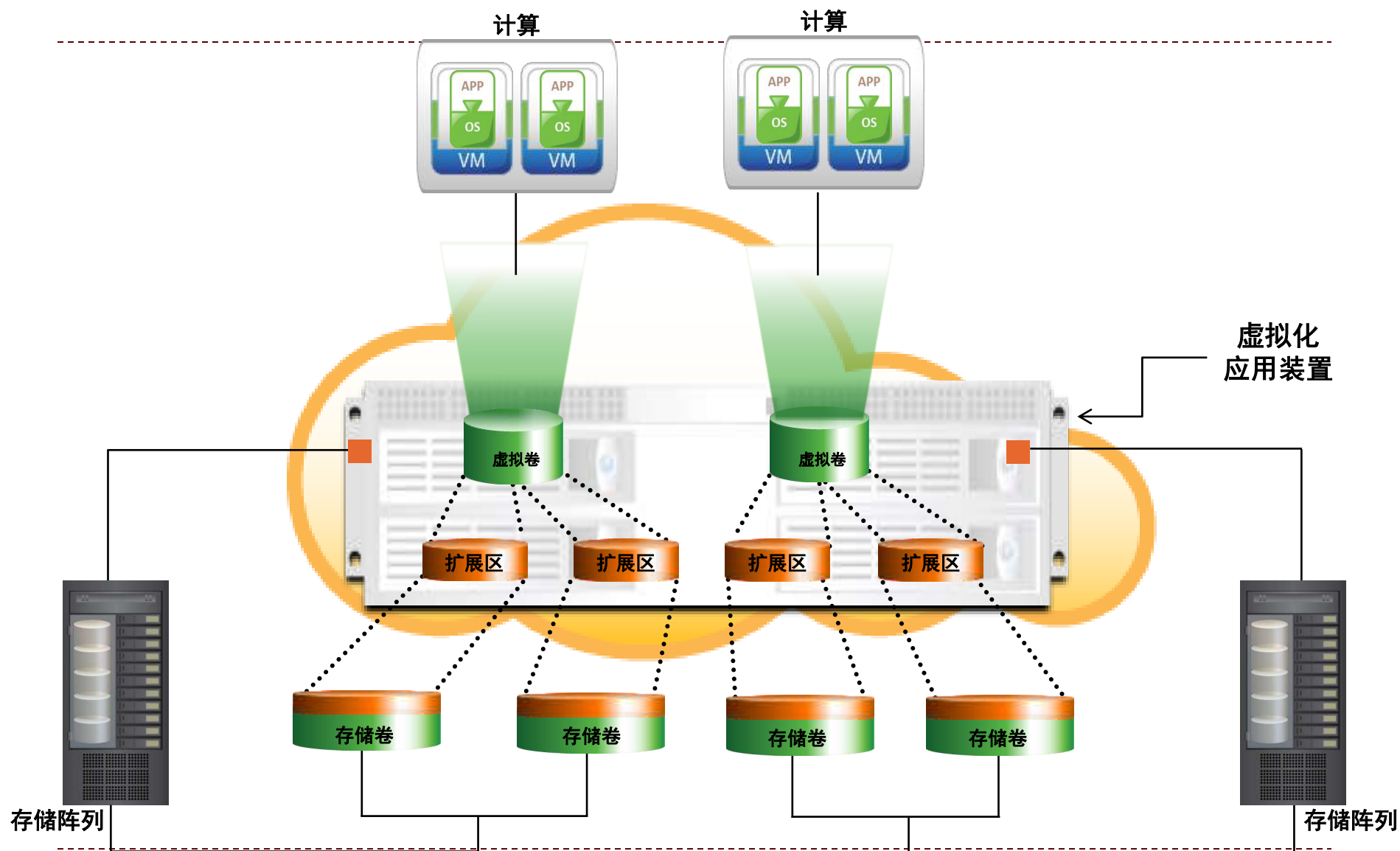
- 基于网络的虚拟化在网络层中嵌入了存储虚拟化智能
- 提供功能以
 - ▶ 创建异构存储资源池
 - ▶ 执行无中断数据迁移
 - ▶ 从单个管理界面管理存储资源池
- 基于网络的存储虚拟化应用于
 - ▶ 数据块级别（存储区域网络）
 - ▶ 文件级别（网络连接存储）

数据块级别的存储虚拟化

- 在存储区域网络中创建一个抽象层位于物理存储资源与提供给计算系统的虚拟卷之间。
- 将计算系统定向到网络层中虚拟化应用装置上的虚拟卷，而非定向到各个存储阵列上的 LUN。
- 允许将一个或多个阵列中的几个 LUN 合并为单个虚拟卷。
- 还可从阵列获取单个大型 LUN，将其分割成较小的虚拟卷。
- 支持存储卷的动态增加、异构存储阵列的整合，以及透明的卷访问。

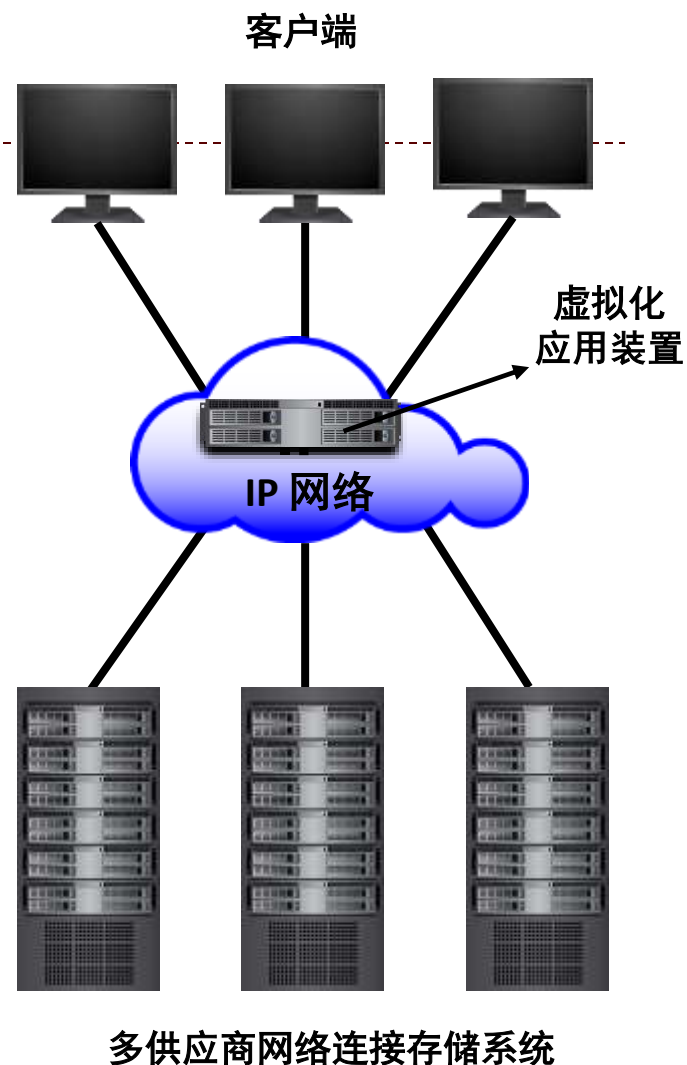


物理到虚拟的卷映射



文件级别的存储虚拟化

- 提供网络连接存储/文件服务器环境的抽象
 - ▶ 消除文件与其位置之间的依赖关系
- 允许在不影响客户端访问的情况下在网络连接存储系统之间移动文件
- 提供机会优化存储利用率
- 使用全局命名空间进行实施



文件级别的存储虚拟化 — 全局命名空间

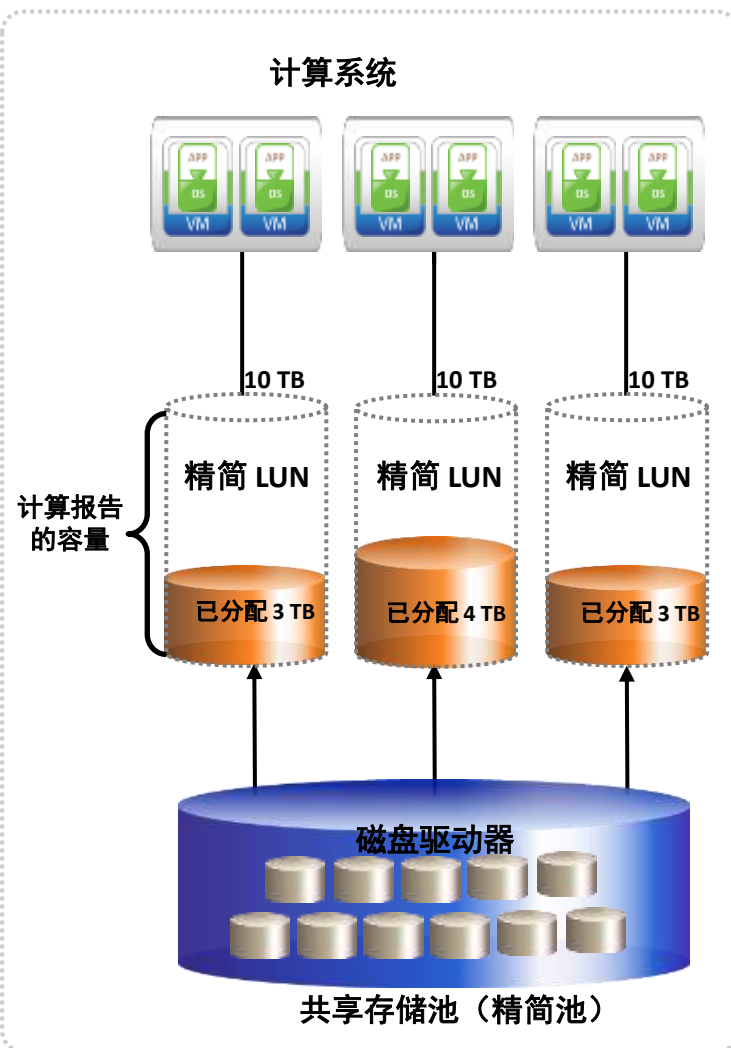
- 允许客户端使用独立于实际物理位置的逻辑名称访问文件
- 将文件的逻辑路径映射到物理路径名称
- 简化文件的访问
 - ▶ 客户端不再需要具有多个装载点来访问位于不同网络连接存储设备上的数据

虚拟资源调配（精简资源调配）

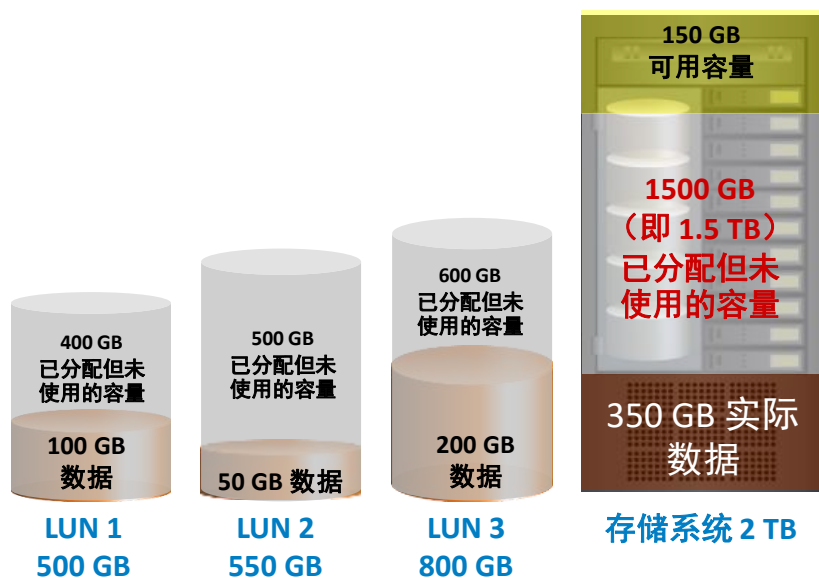
虚拟资源调配（精简资源调配）

它是将 LUN 提供给计算系统的能力，提供的容量大于物理分配给 LUN 的容量。

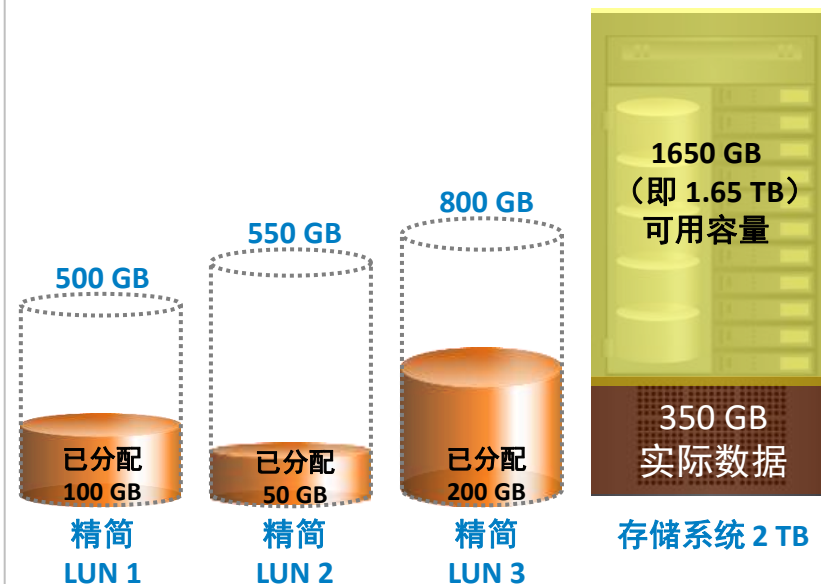
- 从共享存储池按需获取容量，称为精简池
 - ▶ 仅在计算需要时才分配物理存储
 - ▶ 资源调配决策不受当前可用存储的约束
- 可以在以下层实施
 - ▶ 存储层
 - ▶ 计算层 — 对虚拟磁盘进行虚拟资源调配



传统资源调配与 虚拟资源调配



传统资源调配



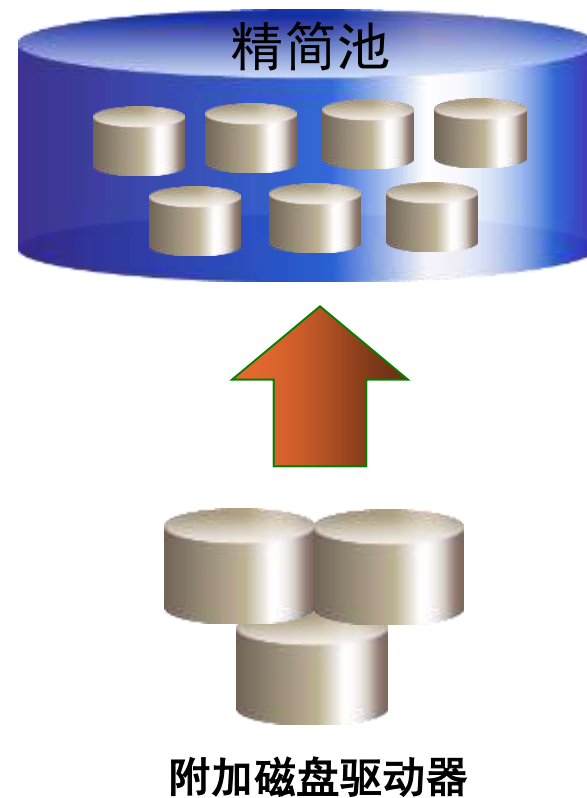
虚拟资源调配

精简 LUN

- 创建时不需要完全分配物理存储的逻辑设备
- 操作系统将其视为传统 LUN
- 从精简池将物理存储分配给精简 LUN
- 从精简池一次分配给精简 LUN 的最小物理存储量称为精简 LUN 扩展区
- 最适合空间效率极为重要的环境

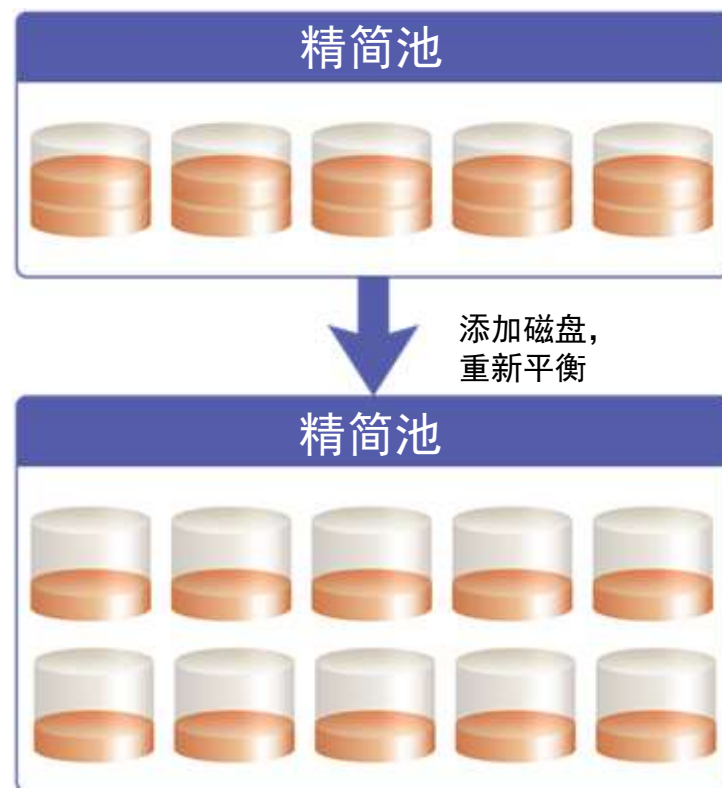
精简池

- 提供精简 LUN 所用实际物理存储的物理驱动器集合
- 在一个存储阵列中可以创建多个池
- 可以动态扩展
 - ▶ 在精简池用于生产时可以将驱动器添加到精简池
- 当销毁精简 LUN 时分配的容量将由池回收



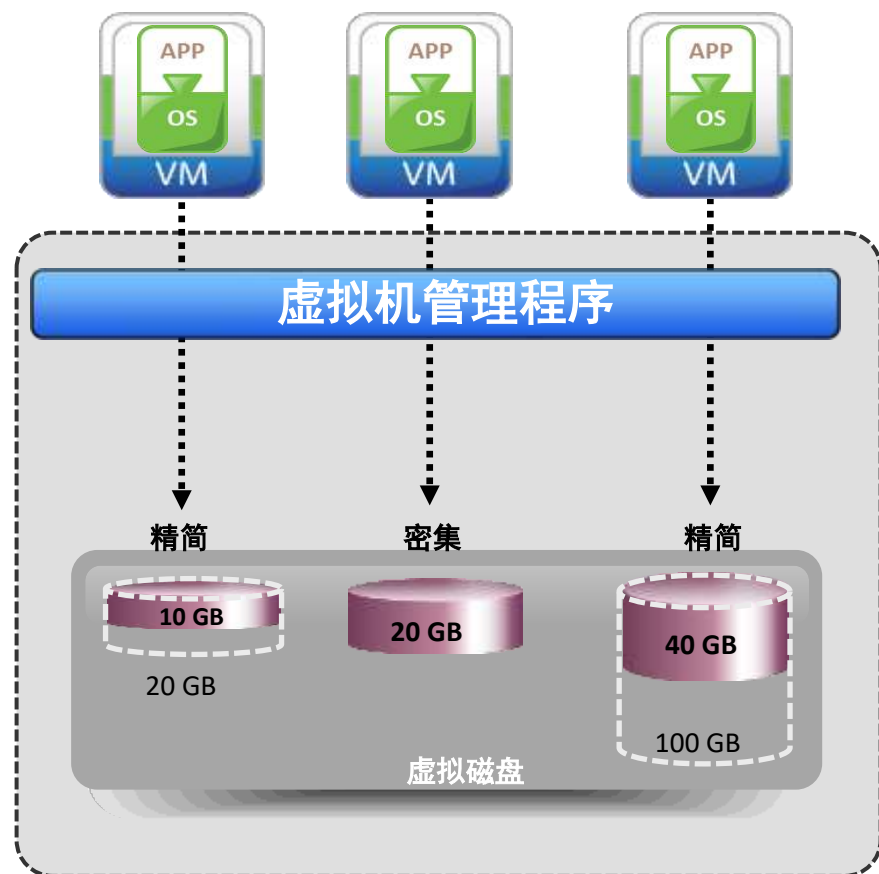
精简池重新平衡

- 添加新的磁盘驱动器时，平衡整个池的物理磁盘驱动器的已用容量
- 在所有磁盘驱动器间重新分配数据

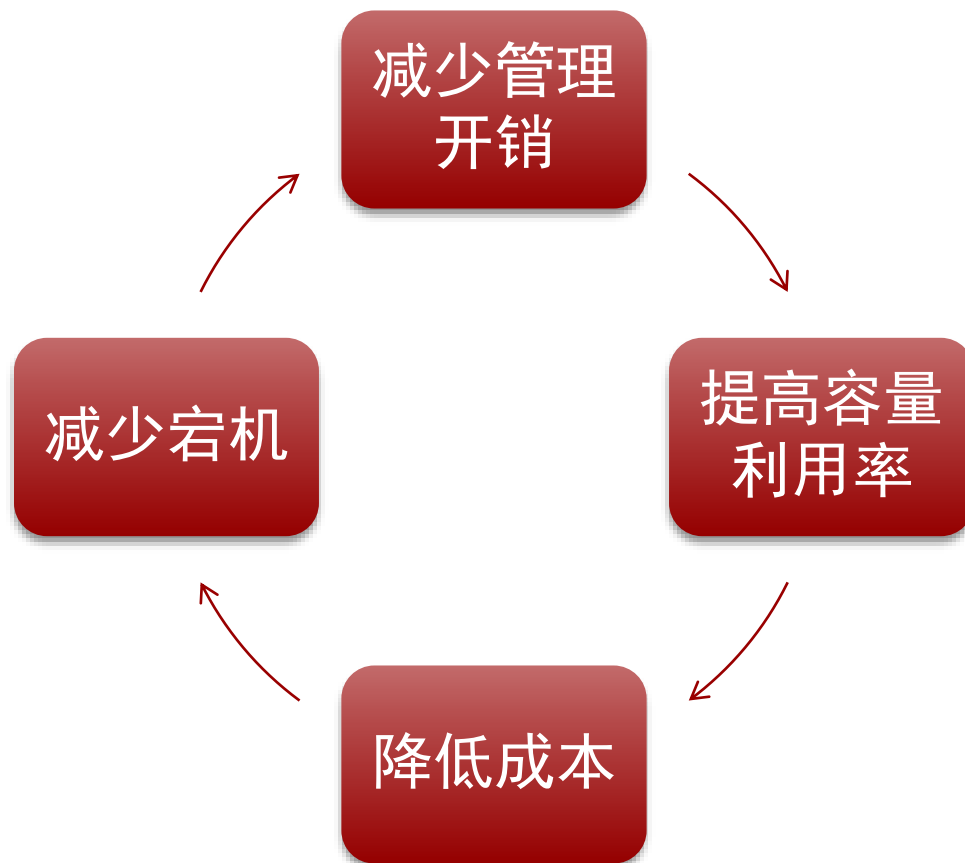


计算级别的虚拟资源调配

- 虚拟机管理程序执行虚拟资源调配来为虚拟机创建虚拟磁盘
 - ▶ 虚拟机始终能看到完整逻辑磁盘大小
- 仅当虚拟机需要存储空间时，虚拟机管理程序才将存储空间分配给虚拟磁盘
 - ▶ 消除虚拟磁盘过度资源调配的需要



虚拟资源调配优势



存储分层

存储分层

建立存储类型的层次结构，并识别候选数据以将它们重新定位到适当的存储类型，以最低的成本满足服务级别要求。

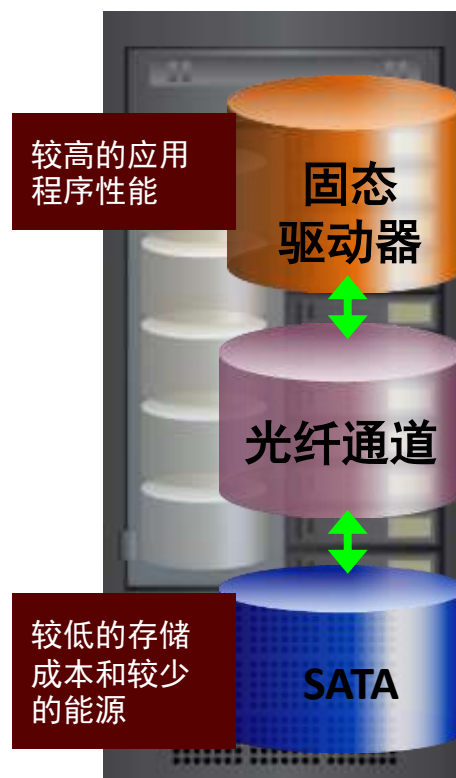
- 每层都针对特定的特征（如性能、可用性或成本）进行优化
- 高效的存储分层要求实施策略
 - ▶ 策略可能基于诸如文件类型、访问频率等参数
- 存储分层的实施
 - ▶ 手动存储分层
 - ▶ 自动存储分层

自动存储分层

- 自动执行存储分层过程
- 允许层之间的无中断数据移动
- 以相同的成本提高应用程序性能，或者以较低的成本提供相同的性能
- 配置数据移动
 - ▶ 在存储阵列内（阵列内）
 - ▶ 在存储阵列之间（阵列间）

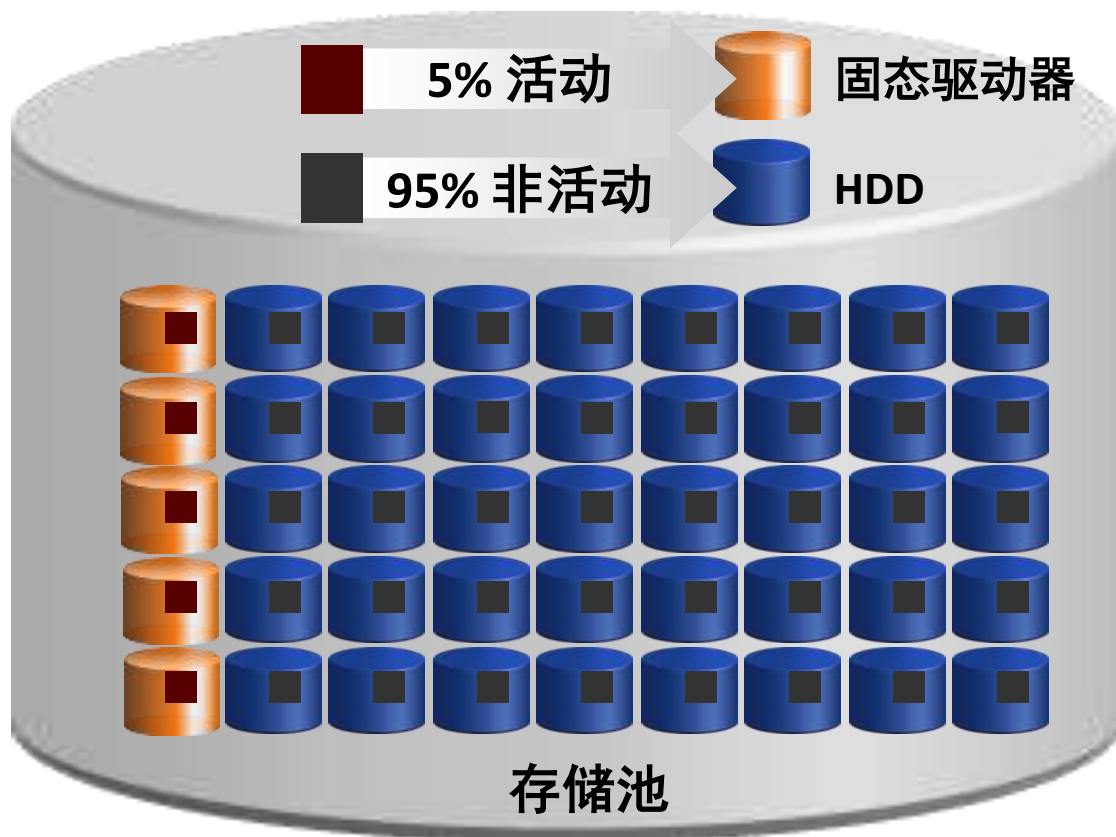
自动存储分层 — 阵列内

- 在阵列内自动执行存储分层过程
- 允许高效使用固态驱动器 (SSD) 和 SATA 驱动器技术
 - ▶ 将活动数据移到高性能的固态驱动器层，将非活动数据移到高容量、低性能的 SATA 驱动器层
- 以子 LUN 级别在层之间执行数据移动
- 利用缓存分层以进一步提高应用程序性能



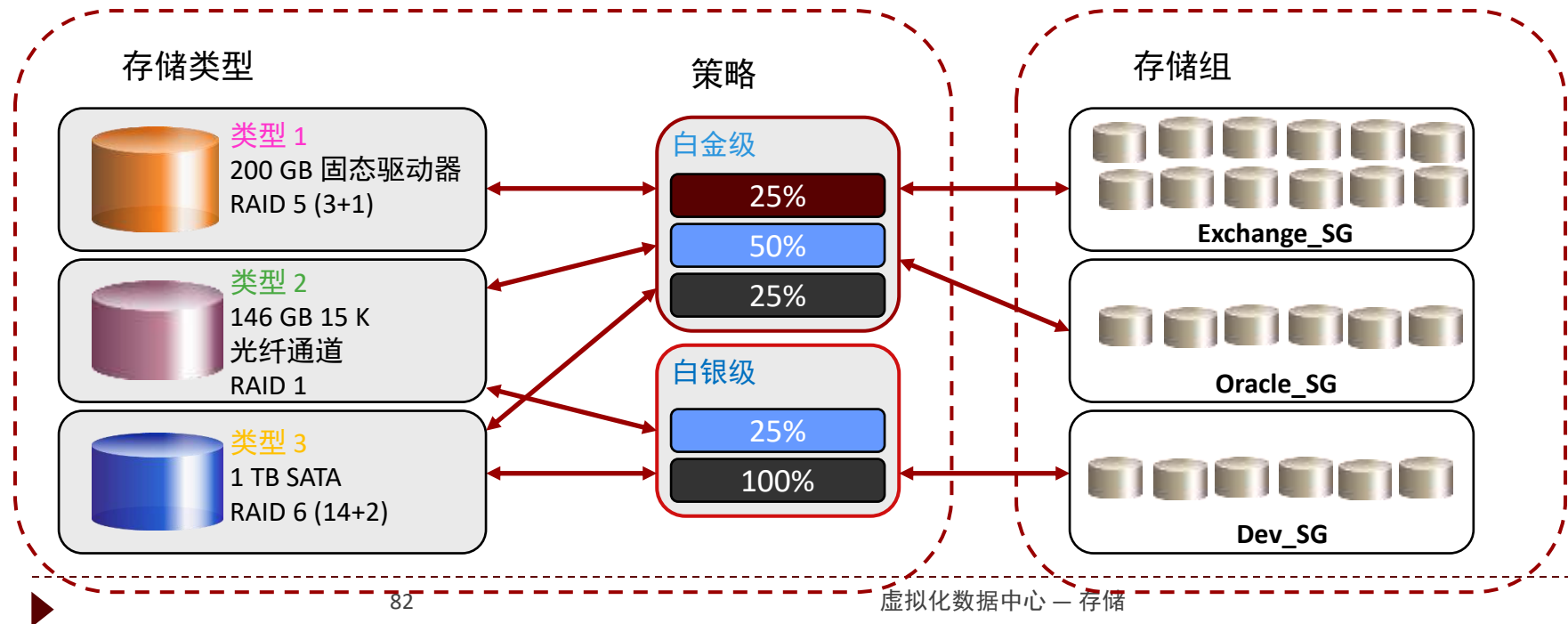
子 LUN 分层

- 通过使用子 LUN 分层，可以将 LUN 分割为较小的片段，然后在该级别进行分层

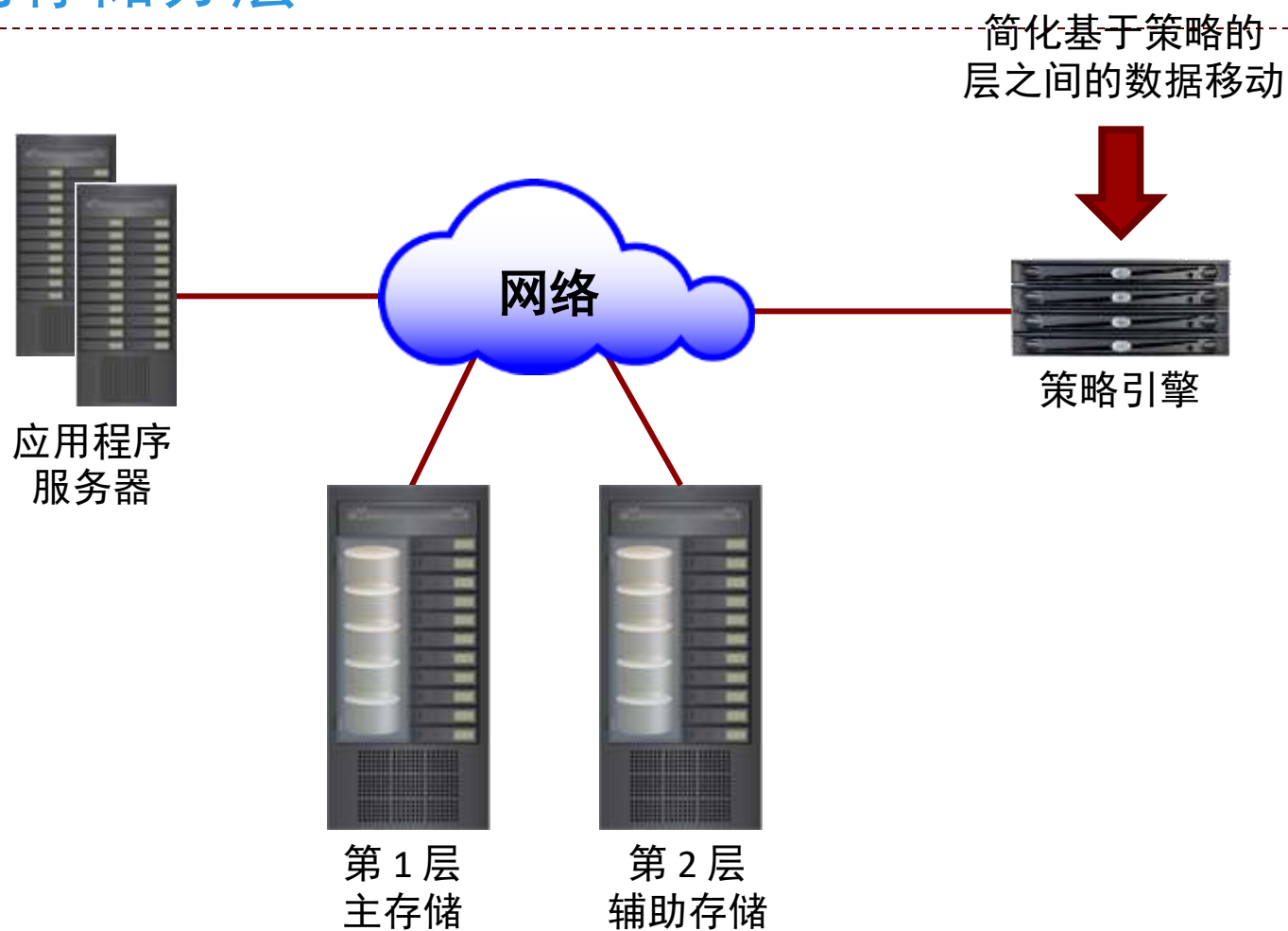


自动存储分层的具体实施

构造块	描述
存储类型	驱动器技术（固态驱动器、光纤通道或 SATA）与 RAID 保护的组合
存储组	一起管理的 LUN 的逻辑集合
策略	管理存储类型之间的数据放置和移动，以满足一个或多个存储组的服务级别



阵列间存储分层



网络虚拟化

网络虚拟化

网络虚拟化

网络虚拟化是指以逻辑方式将物理网络分段或分组为独立的逻辑实体（称为“虚拟网络”），并使它们作为单个或多个独立网络运行。网络虚拟化使多个虚拟网络可以共享网络资源，并且不会相互泄漏信息。

- 使虚拟网络可以共享网络资源
- 虚拟网络对与之连接的节点表现为物理网络
- 针对虚拟网络之间的通信强制执行路由
- 限制管理流量（包括“网络广播”）传播到其他虚拟网络
- 实现虚拟网络中节点的功能分组

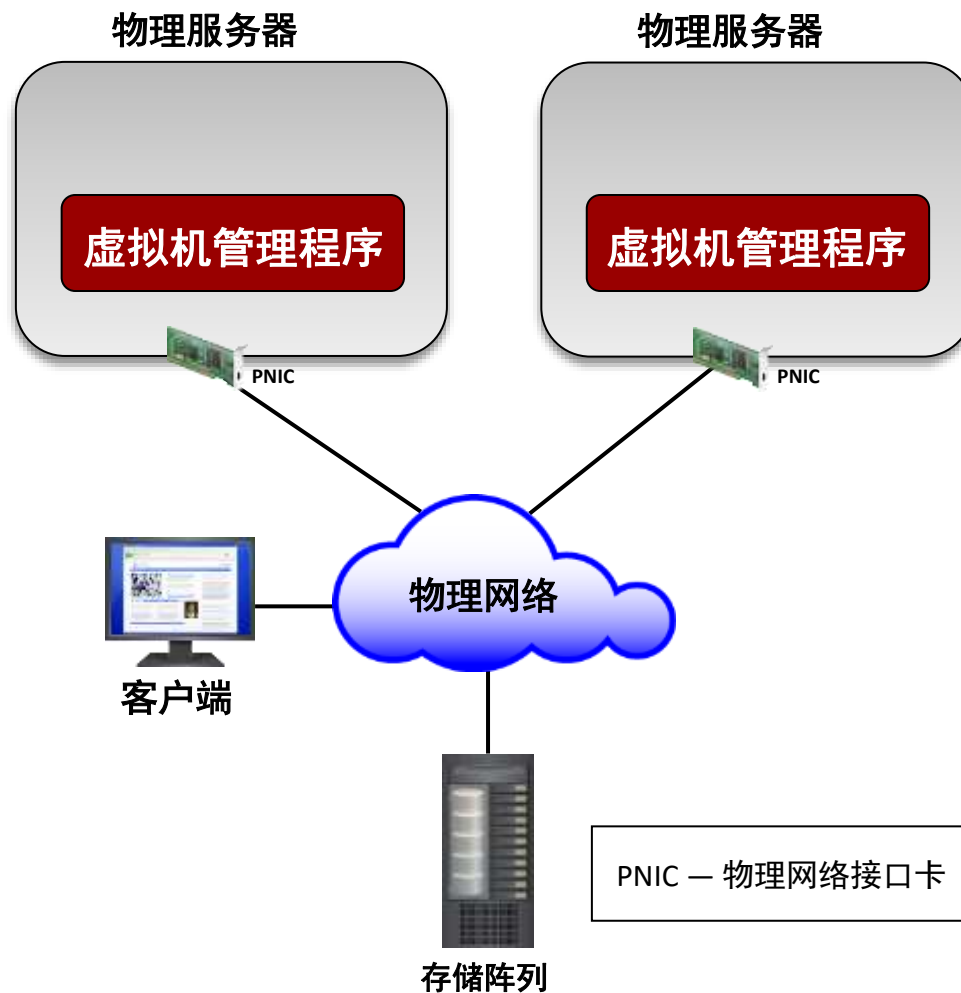


VDC 内的网络虚拟化

- 包括物理网络和虚拟机网络的虚拟化

物理网络

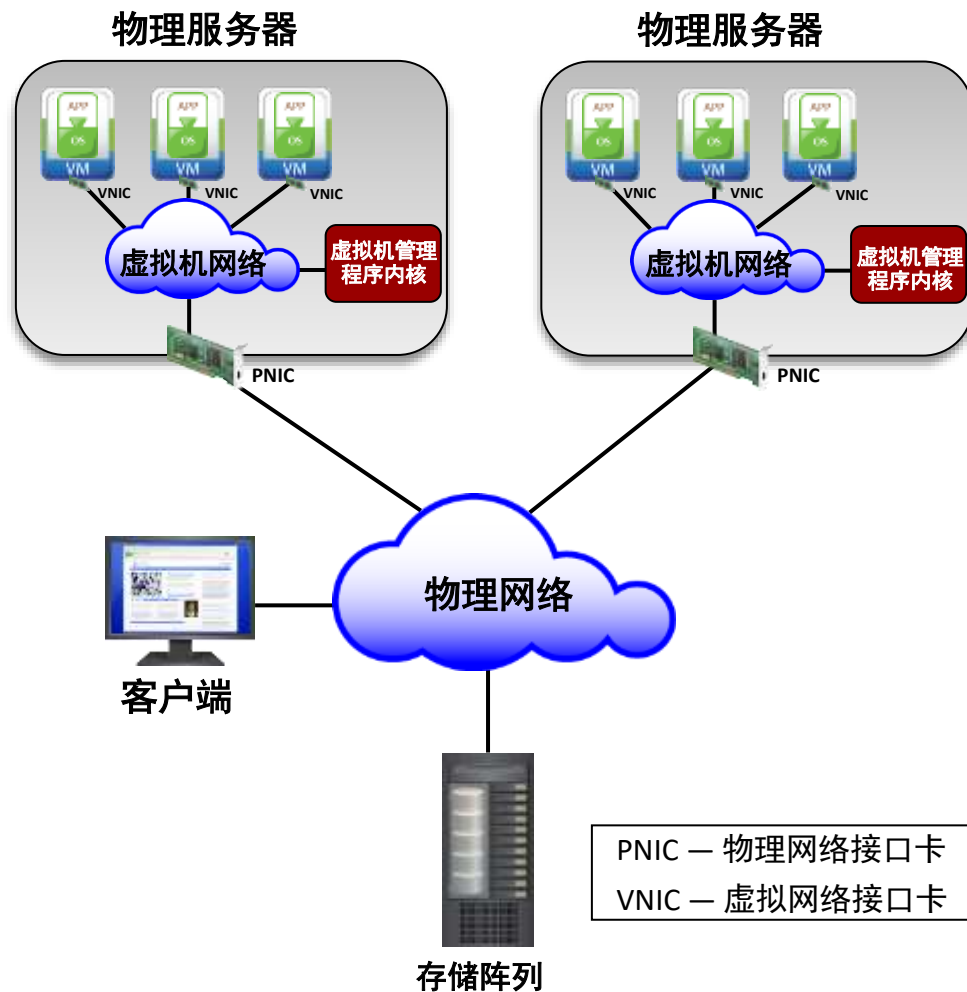
- 由以下物理组件组成：
 - 网络适配器、交换机、路由器、网桥、中继器和集线器
- 提供以下连接：
 - 运行虚拟机管理程序的物理服务器之间的连接
 - 物理服务器与客户端之间的连接
 - 物理服务器与存储系统之间的连接



VDC 内的网络虚拟化（续）

虚拟机网络

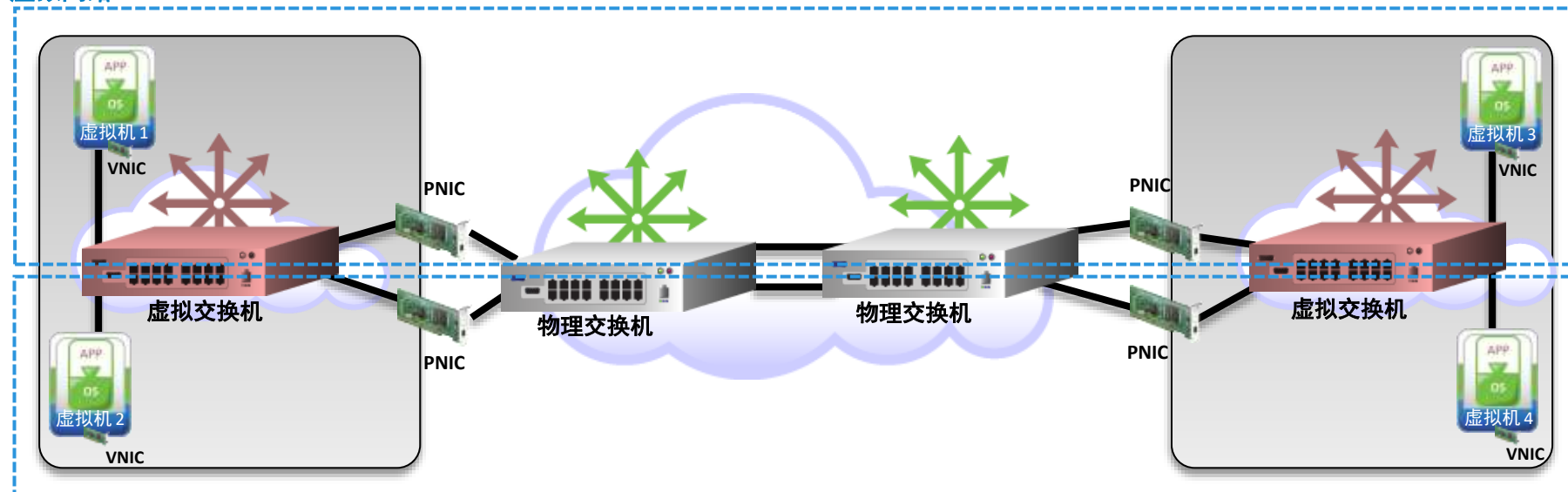
- 驻留在物理服务器中
- 由称为“虚拟交换机”的逻辑交换机组成
- 提供物理服务器中虚拟机之间的连接
- 提供与虚拟机管理程序内核的连接
- 连接到物理网络



VDC 内的网络虚拟化（续）

- 对虚拟机网络和物理网络进行虚拟化可创建虚拟网络，例如：虚拟局域网、虚拟存储区域网络

虚拟网络 1



虚拟网络 2

网络虚拟化的优势

优势	描述
增强安全性	<ul style="list-style-type: none">• 限制从其他虚拟网络对某个虚拟网络中节点的访问• 将一个虚拟网络的敏感数据与另一个虚拟网络隔离
提高性能	<ul style="list-style-type: none">• 限制网络广播并提高虚拟网络性能
提高可管理性	<ul style="list-style-type: none">• 允许使用管理软件从一个集中化管理工作站配置虚拟网络• 简化节点的分组和重新分组
提高利用率	<ul style="list-style-type: none">• 使多个虚拟网络可以共享同一个物理网络，从而提高网络资源的利用率• 减少针对不同节点组设置单独的物理网络的需求

网络虚拟化工具

- 物理交换机操作系统 (OS)
 - ▶ 操作系统必须具有网络虚拟化功能
- 虚拟机管理程序
 - ▶ 使用内置的联网和网络虚拟化功能
 - ▶▶ 创建虚拟交换机并配置虚拟网络
 - ▶ 或者，使用第三方软件提供联网和网络虚拟化功能
 - ▶▶ 第三方软件安装在虚拟机管理程序上
 - ▶▶ 第三方软件将替换虚拟机管理程序的固有联网功能

VDC 网络基础架构的组件

- VDC 网络基础架构同时包括虚拟网络组件和物理网络组件
 - ▶ 这些组件相互连接以传输网络流量

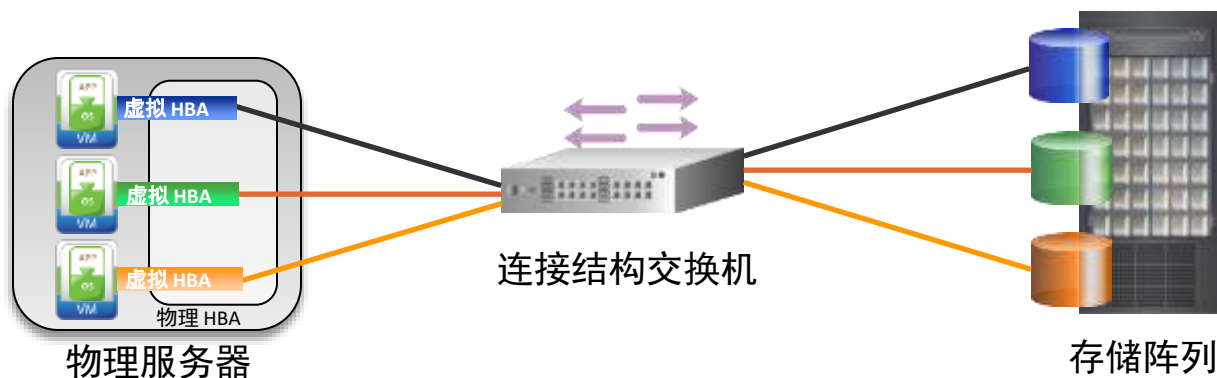
组件	描述
虚拟网络接口卡	<ul style="list-style-type: none">• 将虚拟机连接到虚拟机网络• 与虚拟机网络之间双向传输虚拟机流量
虚拟 HBA	<ul style="list-style-type: none">• 使虚拟机可以访问为其分配的光纤通道原始设备映射磁盘/LUN
虚拟交换机	<ul style="list-style-type: none">• 构成虚拟机网络的以太网交换机• 提供与虚拟网络接口卡的连接并转发虚拟机流量• 提供与虚拟机管理程序内核的连接并定向虚拟机管理程序流量：管理、存储、虚拟机迁移
物理适配器：网络接口卡 (NIC)、HBA、聚合网络适配器	<ul style="list-style-type: none">• 将物理服务器连接到物理网络• 与物理网络之间双向转发虚拟机流量和虚拟机管理程序流量
物理交换机、路由器	<ul style="list-style-type: none">• 构成支持以太网/光纤通道/iSCSI/FCoE 的物理网络• 提供以下连接：物理服务器之间的连接、物理服务器与存储系统之间的连接，以及物理服务器与客户端之间的连接

虚拟网络组件：虚拟网络接口卡

- 将虚拟机连接到虚拟交换机
- 将以太网帧转发到虚拟交换机
- 具有唯一的 MAC 地址和 IP 地址
- 支持与物理网络接口卡类似的以太网标准

虚拟网络组件：虚拟 HBA

- 使虚拟机可以访问为其分配的光纤通道原始设备映射磁盘/LUN
- 使用 N 端口 ID 虚拟化 (NPIV) 技术进行配置
 - ▶ 由单个物理光纤通道 HBA 或聚合网络适配器端口（N 端口）提供多个虚拟 N 端口（分别具有自己的全球通用名称）的功能
 - ▶ 由虚拟 N 端口提供虚拟 HBA 的功能
- 虚拟机管理程序内核利用 NPIV 对虚拟 N 端口进行实例化
 - ▶ 将虚拟 N 端口分配给虚拟机
- 实现虚拟机级别的分区和 LUN 掩蔽

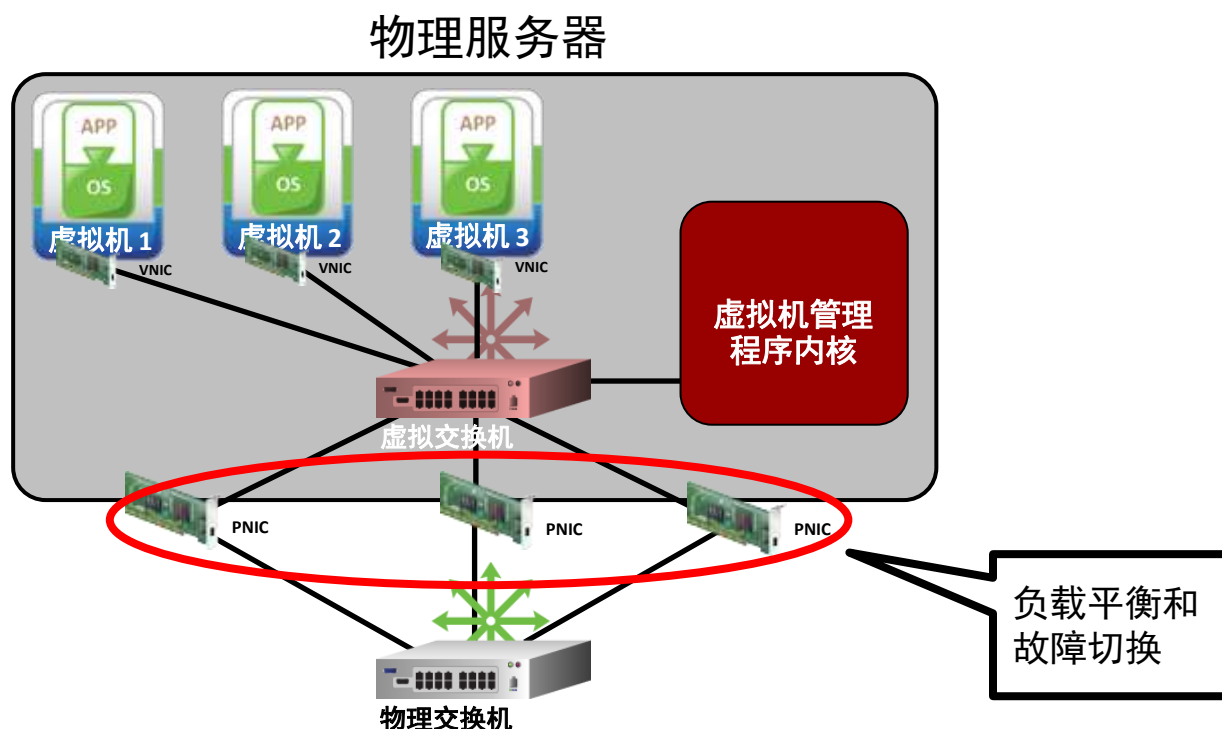


虚拟网络组件：虚拟交换机

- 支持以太网协议的逻辑 OSI 第 2 层交换机
- 驻留在物理服务器中
- 使用虚拟机管理程序进行创建和配置
- 虚拟交换机为虚拟机和虚拟机管理程序内核提供流量管理
- 保存用于帧转发的 MAC 地址表
- 与虚拟机和虚拟机管理程序内核之间双向转发网络流量
 - ▶ 物理服务器中的虚拟机到虚拟机
 - ▶ 虚拟机到物理网络
 - ▶ 虚拟机管理程序内核：IP 存储、虚拟机迁移和管理

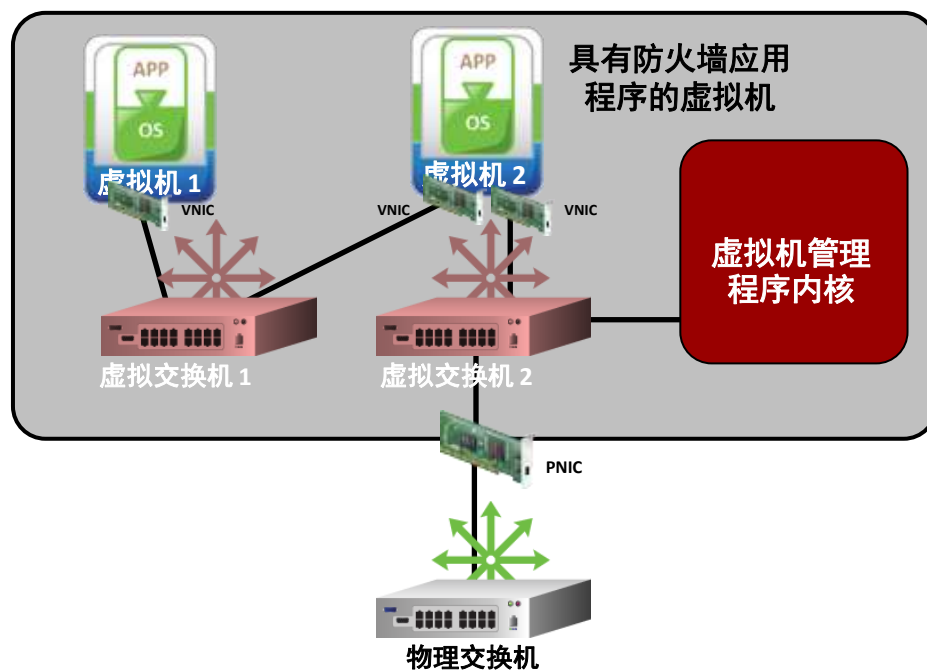
虚拟网络组件：虚拟交换机（续）

- 可连接到多个物理网络接口卡
 - ▶ 与多个网络接口卡连接可执行负载平衡和故障切换



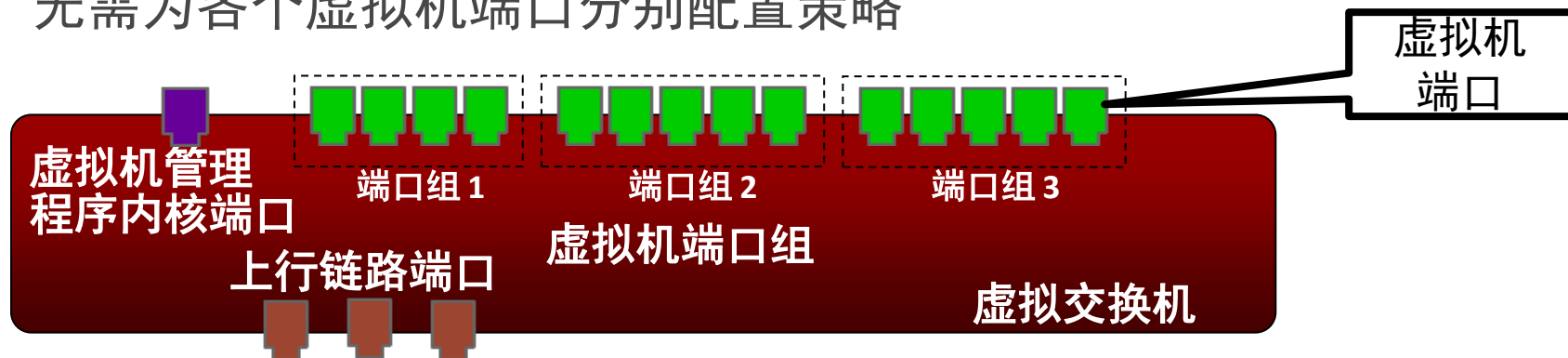
虚拟网络组件：虚拟交换机（续）

- 可以不连接任何物理网络接口卡
 - ▶ 如果虚拟交换机未连接任何物理网络接口卡，则会在物理服务器内定向虚拟机流量



虚拟交换机：端口和端口组

- 端口类型
 - ▶ 虚拟机管理程序内核端口：为虚拟机管理程序内核提供连接
 - ▶ 虚拟机端口：为虚拟网络接口卡提供连接
 - ▶ 上行链路端口：为物理网络接口卡提供连接
- 虚拟机端口组：将统一网络策略设置应用于一组虚拟机端口的机制
 - ▶ 策略示例：安全性、负载平衡和跨 PNIC 故障切换
- 连接到同一个虚拟机端口组的虚拟机共享通用配置
 - ▶ 无需为各个虚拟机端口分别配置策略

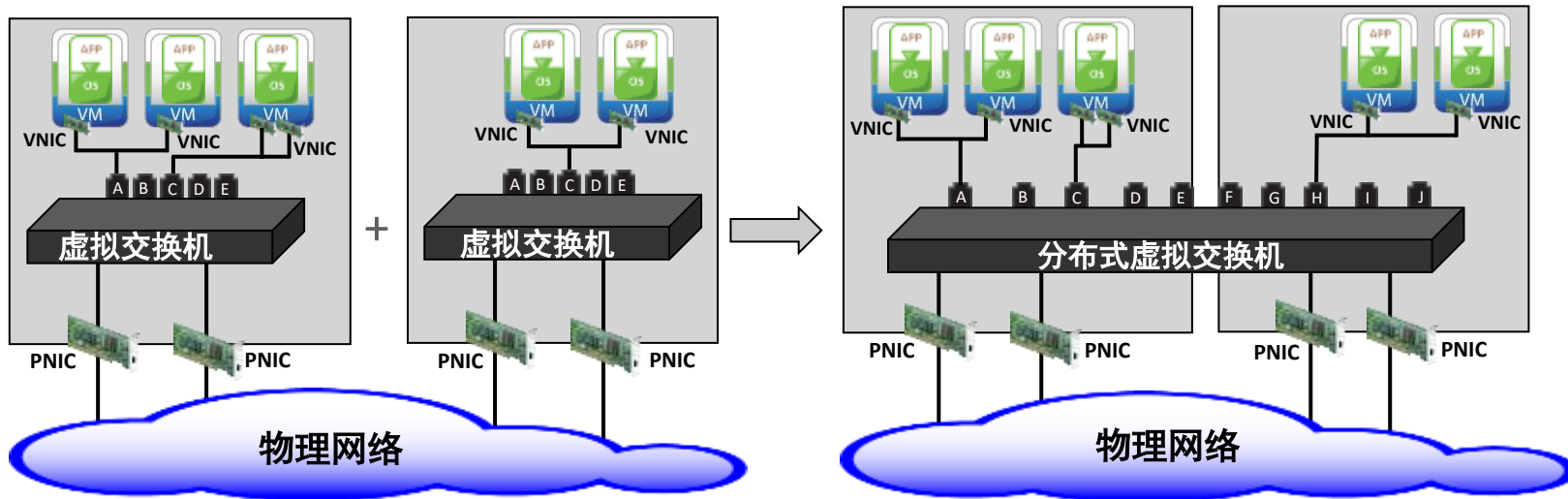


分布式虚拟交换机

- 跨多个物理服务器分布的多个虚拟交换机的聚合

优势

- 实现集中式虚拟机网络管理
- 在虚拟机迁移过程中保持网络策略



物理网络组件：网络接口卡 (NIC)

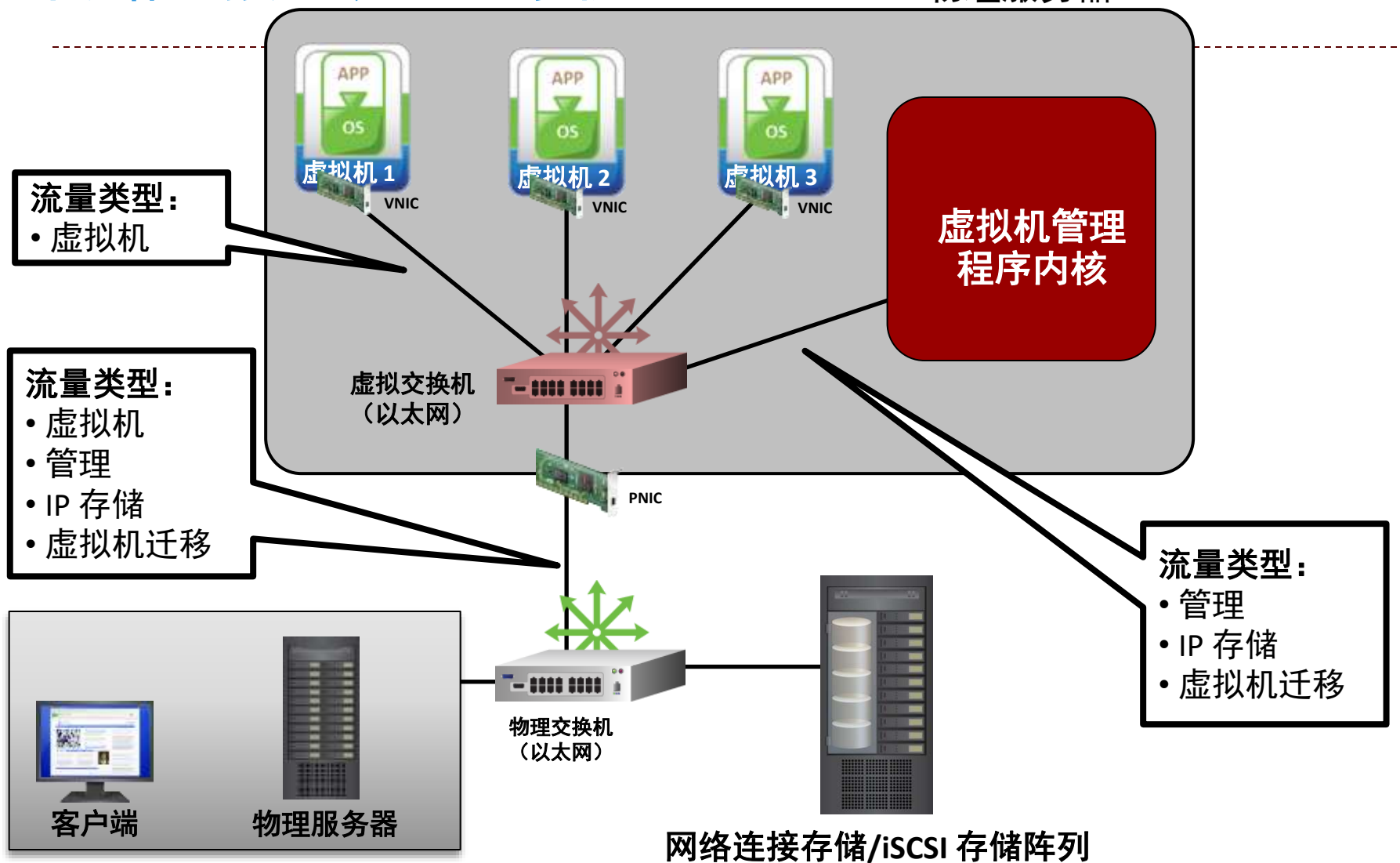
- 物理网络接口卡用作虚拟交换机与物理以太网交换机之间的交换机间链路
 - ▶ 传输虚拟机流量和虚拟机管理程序内核流量
- 物理网络接口卡无法通过网络寻址
 - ▶ 未分配 IP 地址（禁止 OSI 第 3 层访问）
 - ▶ MAC 地址不可用（禁止 OSI 第 2 层访问）
- 虚拟网络接口卡和虚拟机管理程序内核可通过网络寻址
 - ▶ 具有自己的 MAC 地址和 IP 地址
 - ▶▶ 用作以太网帧中的源地址
- 以太网帧在无修改的情况下通过物理网络接口卡进行传输

物理网络组件：HBA 和聚合网络适配器

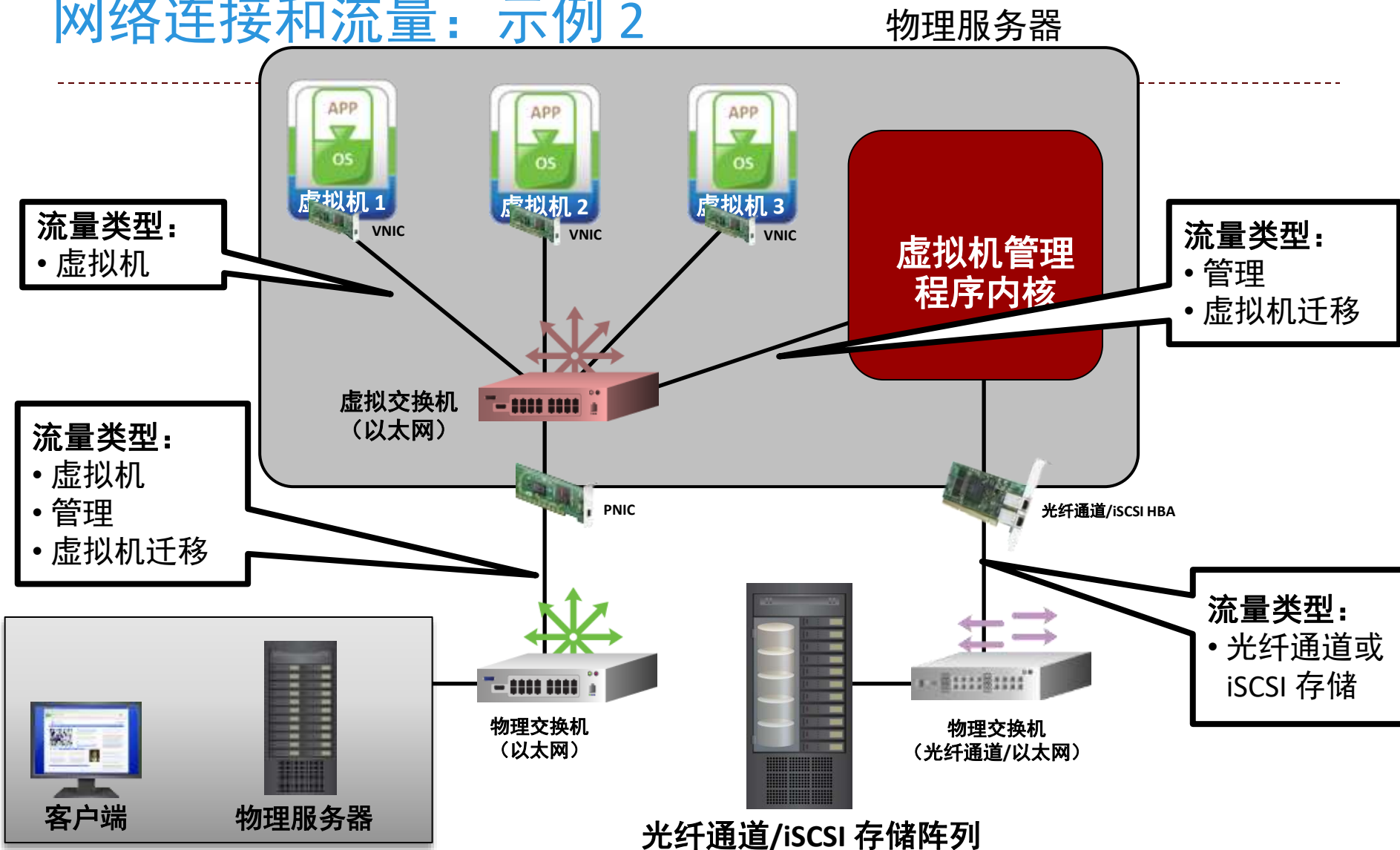
适配器类型	描述
iSCSI HBA	<ul style="list-style-type: none">• 将虚拟机管理程序存储 I/O (SCSI I/O) 传输到 iSCSI 存储系统• 具有内置 iSCSI 启动器• 将 SCSI I/O 封装到 iSCSI 帧中，然后将 iSCSI 帧封装到以太网帧中• 使用自己的 MAC 地址和 IP 地址通过以太网传输以太网帧• 使虚拟机管理程序无需进行 iSCSI 处理（SCSI 到 iSCSI）
光纤通道 HBA	<ul style="list-style-type: none">• 将虚拟机管理程序存储 I/O (SCSI I/O) 传输到光纤通道存储系统• 将 SCSI 数据封装到光纤通道帧中• 使用自己的光纤通道地址通过光纤通道网络传输帧
聚合网络适配器	<ul style="list-style-type: none">• 虚拟机管理程序识别为光纤通道 HBA 和网络接口卡<ul style="list-style-type: none">▶ 网络接口卡：用作虚拟交换机与物理交换机之间的链路▶ 光纤通道 HBA：为虚拟机管理程序提供对光纤通道存储的访问

网络连接和流量：示例 1

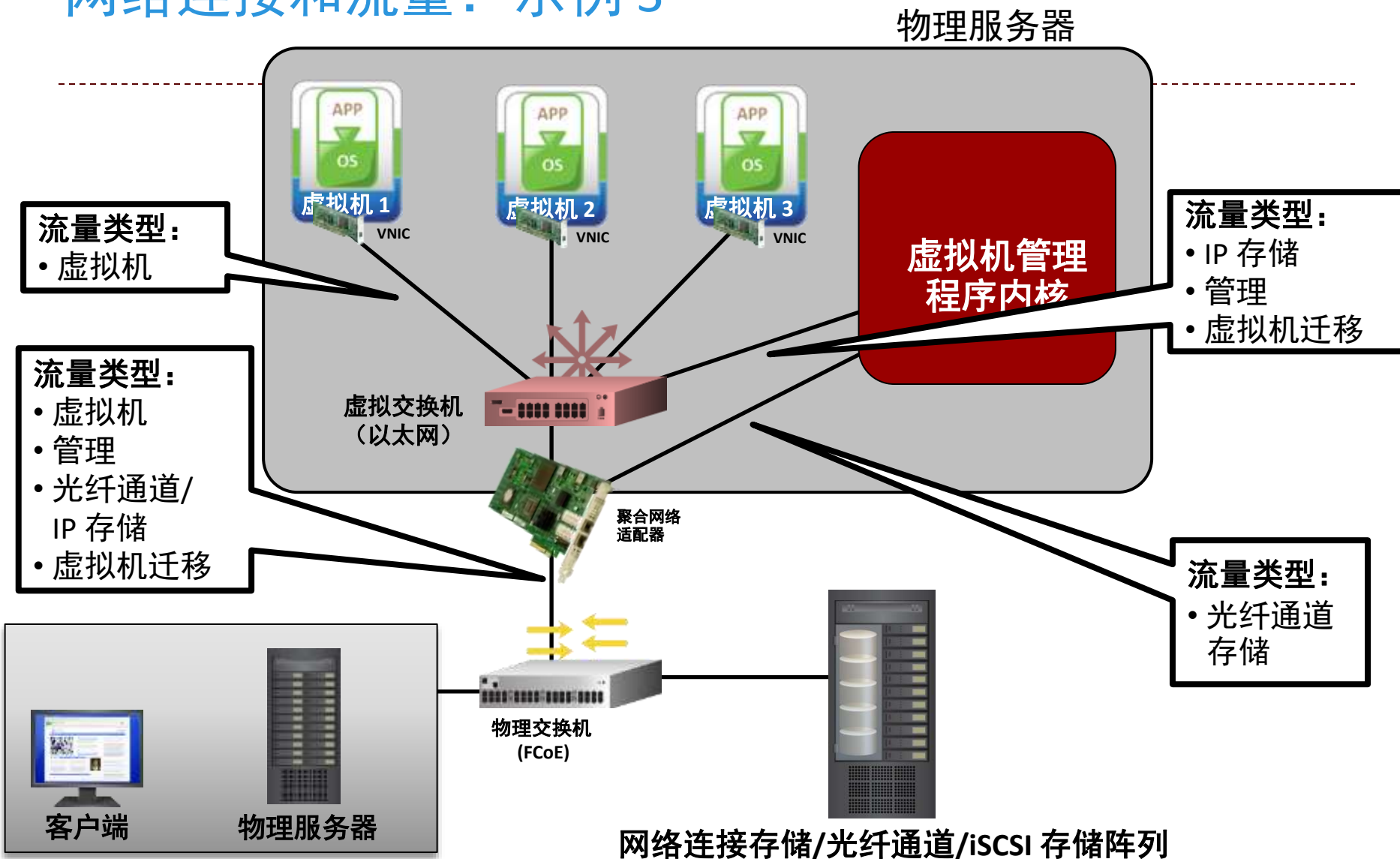
物理服务器



网络连接和流量：示例 2



网络连接和流量：示例 3



虚拟局域网 (VLAN)

虚拟局域网

在某个局域网中或跨多个由物理交换机和虚拟交换机组成的局域网创建的逻辑网络，可实现一组节点之间的通信（不考虑节点在网络中的位置）。

优势

- 控制广播活动并提高网络性能
- 简化管理
- 提高安全性级别
- 提高交换机的利用率并减少资本成本

配置虚拟局域网

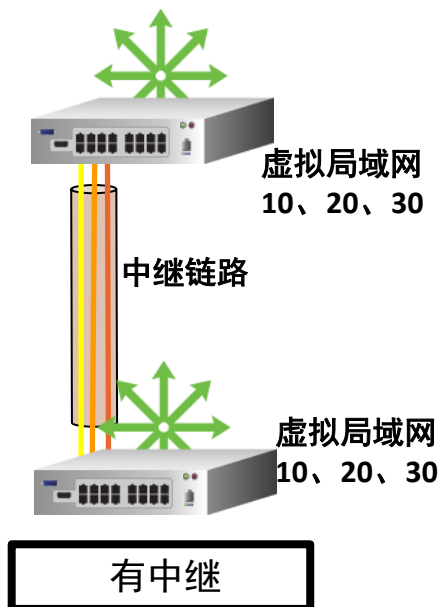
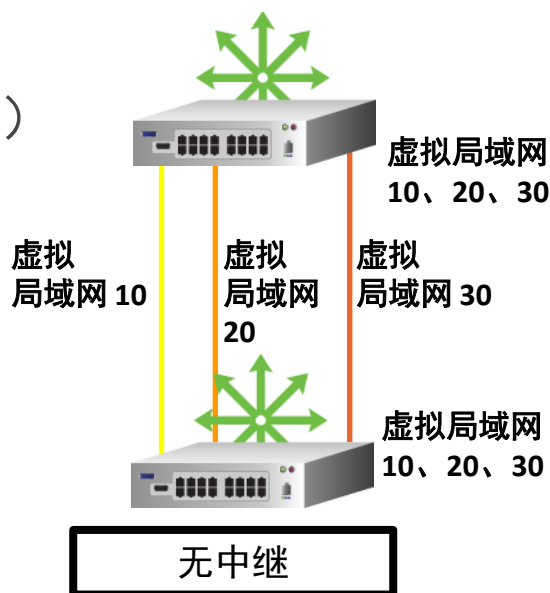
- 在物理交换机上定义虚拟局域网 ID
 - ▶ 每个虚拟局域网由一个唯一的编号标识：虚拟局域网 ID
- 从虚拟机管理程序的内置虚拟局域网 ID 池中选择必要的虚拟局域网 ID
 - ▶ 对于虚拟交换机是必需的
- 将虚拟局域网 ID 分配给物理交换机端口和虚拟交换机端口
 - ▶ 将交换机端口纳入虚拟局域网
 - ▶ 允许将交换机端口分组到虚拟局域网中

虚拟局域网中继

虚拟局域网中继

这种技术使来自多个虚拟局域网的流量可以通过单一网络连接

- 使用单一连接（中继链路）承载多重虚拟局域网流量
- 通过单个端口（中继端口）发送/接收多重虚拟局域网流量
- 将中继端口纳入所有虚拟局域网
- 通过标记以太网帧实现虚拟局域网中继



虚拟局域网中继的优势

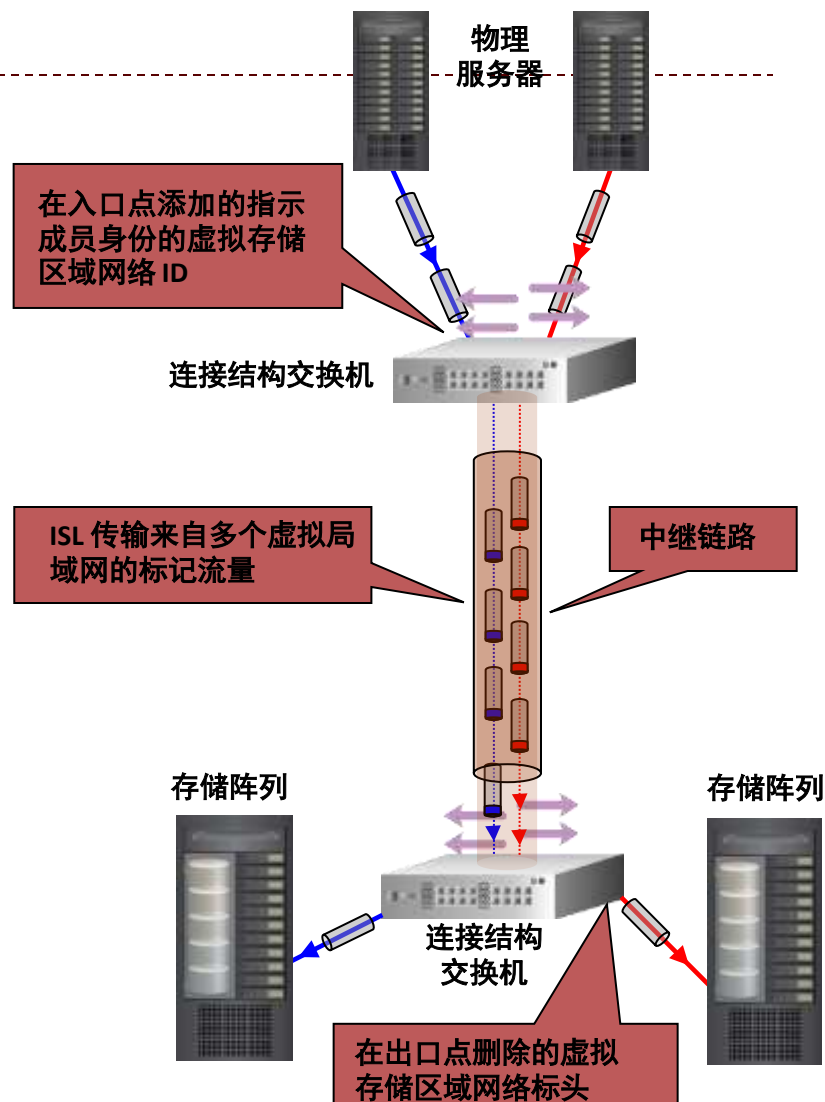
- 无需针对每个虚拟局域网使用专用网络链路
- 减少存在多个虚拟局域网时的设备间链路
 - ▶ 减少虚拟网络接口卡、存储端口和交换机端口的数量
 - ▶ 降低管理复杂性

虚拟存储区域网络 (VSAN)

虚拟存储区域网络

光纤通道 SAN 中的逻辑连接结构，可实现一组节点之间的通信（不考虑节点在连接结构中的物理位置）。

- 虚拟存储区域网络具有自己的连接结构服务（名称服务器、分区）、配置和光纤通道地址集
- 虚拟存储区域网络的优势与虚拟局域网类似
- 利用虚拟存储区域网络标记，多重虚拟存储区域网络流量可以通过同一个中继链路



网络流量管理的要求

- 负载均衡
 - ▶ 将工作负载分布到多个 IT 资源
 - ▶ 防止资源利用率过高/过低，并优化性能
- 基于策略的管理
 - ▶ 允许使用跨虚拟机和网络链路分布流量的策略
 - ▶ 允许使用跨网络链路故障切换流量的策略
- 无争用资源共享
 - ▶ 当来自多个虚拟网络的流量共享物理网络资源时，提供有保证的服务级别
 - ▶ 设置为不同类型的流量分配带宽的优先级

主要网络流量管理技术

1. 平衡客户端工作负载：基于硬件
2. 平衡客户端工作负载：基于软件
3. 风暴控制
4. 网络接口卡绑定
5. “限制”和“共享”
6. 流量调整
7. 多路径

技术 1 — 平衡客户端工作负载：基于硬件

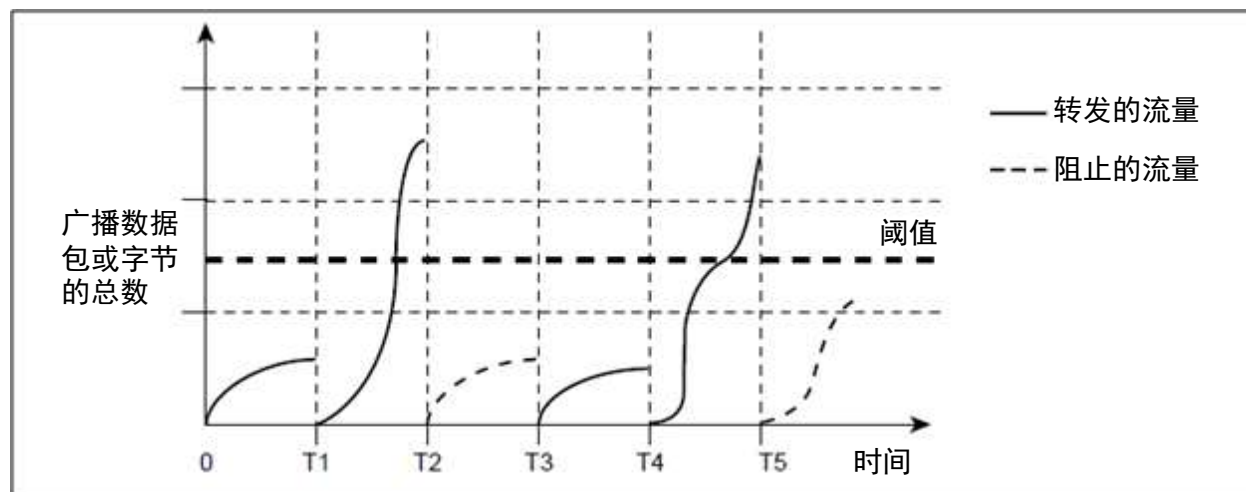
- 设备（物理交换机/路由器）将客户端流量分布到多个服务器 — 物理计算机或虚拟机
- 客户端使用负载均衡设备的 IP 地址（虚拟）发送请求
- 负载均衡设备决定转发请求的目标
- 决策通常由负载均衡策略控制，例如：循环、加权循环、最少连接
 - 循环策略使客户端连接跨服务器循环。
 - 加权循环允许管理员为每个服务器定义性能权重。权重值较高的服务器在循环过程中将获得较大的连接百分比。
 - 最少连接策略使所有服务器保持相同的连接数。随着时间的推移，能够更快处理连接的服务器将获得更多连接。

技术 2 — 平衡客户端工作负载：基于软件

- 由物理计算机或虚拟机上运行的软件执行
 - ▶ 示例：DNS 服务器负载均衡
 - ▶▶ 允许一个域名具有多个 IP 地址
 - ▶▶ 以循环方式将域名映射到不同的 IP 地址
 - ▶▶ 允许访问某个域名的客户端向不同的服务器发送请求
 - ▶ 示例：Microsoft 网络负载均衡
 - ▶▶ 群集中每个服务器上的特殊驱动程序，用于平衡客户端的工作负载
 - ▶▶ 驱动程序对所有客户端显示同一个 IP 地址（虚拟）— 每个 IP 数据包到达每个服务器
 - ▶▶ 驱动程序将每个请求映射到一个特定的服务器 — 群集中的其他服务器将丢弃该请求

技术 3 — 风暴控制

- 防止风暴对常规局域网/虚拟局域网流量产生影响
 - 风暴：局域网/虚拟局域网中的帧的泛滥，会产生过多流量并降低网络性能
- 计算 1 秒内指定类型的帧的数量，并与阈值进行比较
- 如果达到阈值，交换机端口将阻止流量，并在下一个时间间隔内丢弃后续帧



来源：《Catalyst 3550 Multilayer Switch Software Configuration Guide》— Cisco Systems, Inc.

技术 4 — 网络接口卡绑定

- 对连接到某个虚拟交换机的物理网络接口卡进行逻辑分组
 - ▶ 创建网络接口卡组，其成员可以处于活动状态和备用状态
 - ▶ 跨活动网络接口卡组成员平衡流量负载
 - ▶ 在网络接口卡/链路发生故障时提供故障切换
 - ▶ 允许在虚拟交换机或端口组中关联负载平衡和故障切换策略

技术 5 — “限制” 和 “共享”

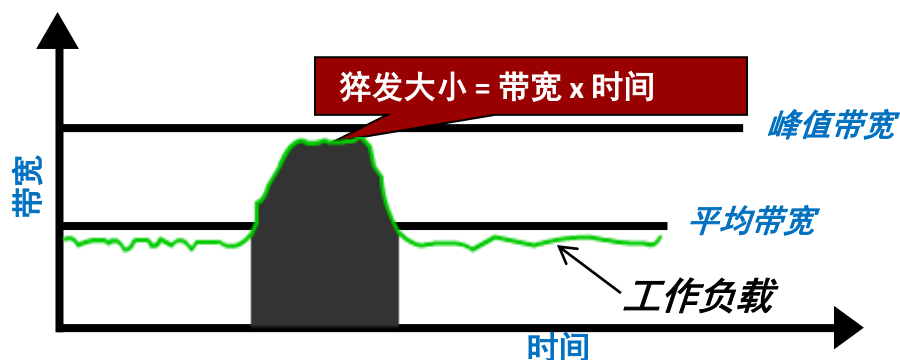
- 分布式虚拟交换机中的可配置参数
- 配置用于控制竞争某个物理网络接口卡或网络接口卡组的不同类型的网络流量
- 确保业务关键型应用程序获得所需的带宽

可配置参数	描述
限制	<ul style="list-style-type: none">• 设置每种流量类型的最大带宽限制<ul style="list-style-type: none">▸ 流量类型不会超过限制• 以 Mbps 为单位指定• 适用于网络接口卡组
共享	<ul style="list-style-type: none">• 指定为不同流量类型分配带宽的优先级• 以数字形式指定• 适用于物理网络接口卡

技术 6 — 流量调整

- 在虚拟/分布式虚拟交换机或端口组中控制网络带宽
- 防止业务关键型应用程序流量受到非关键型流量的影响

参数	描述
平均带宽	<ul style="list-style-type: none">• 一段时间内允许的数据传输率• 交换机端口的工作负载可能会间歇性超过平均 带宽<ul style="list-style-type: none">• 猝发：当工作负载超过平均带宽时，便称为猝发
峰值带宽	<ul style="list-style-type: none">• 在没有排队/丢弃帧的情况下的最大数据传输率
猝发大小	<ul style="list-style-type: none">• 在猝发中允许传输的数据的最大量• 猝发大小= “带宽” × “时间”• 猝发中的带宽最高可达峰值带宽



示例：

平均带宽 = 1 Kbps，峰值带宽 = 4 Kbps，
猝发大小 = 3 Kb

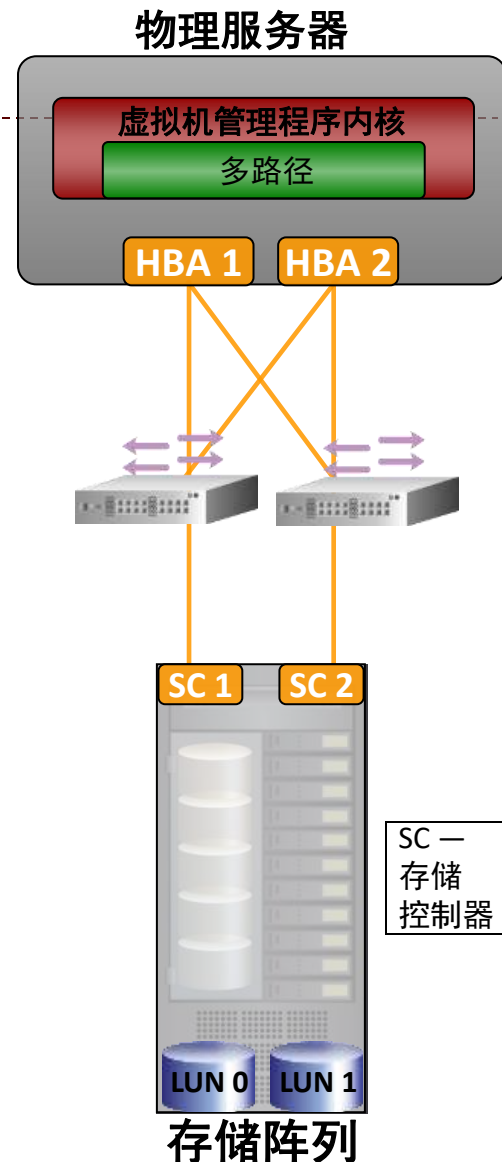
- 数据速率为 3 Kbps 的猝发可持续 1 秒，
- 数据速率为 1.5 Kbps 的猝发可持续 2 秒，
依此类推

技术 7 — 多路径

多路径

利用这种技术，物理服务器可以使用多个物理路径在物理服务器与存储系统上的 LUN 之间传输数据。

- 在虚拟机管理程序中内置或由第三方供应商提供
- 识别指向 LUN 的替代 I/O 路径并实现故障切换
- 通过将 I/O 分布到所有可用路径，执行负载平衡



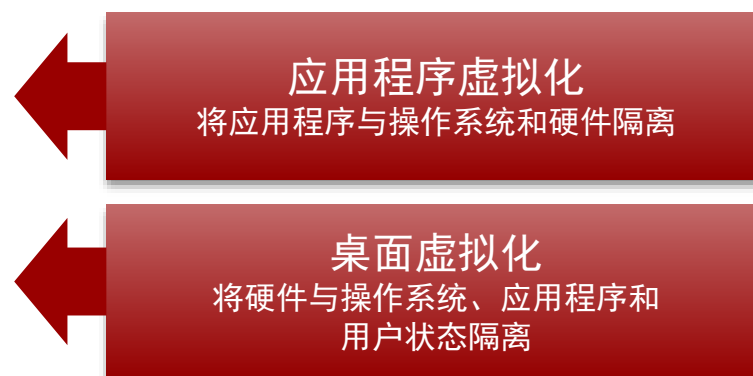
桌面虚拟化

桌面和应用程序虚拟化概述

各层之间的紧密依赖关系

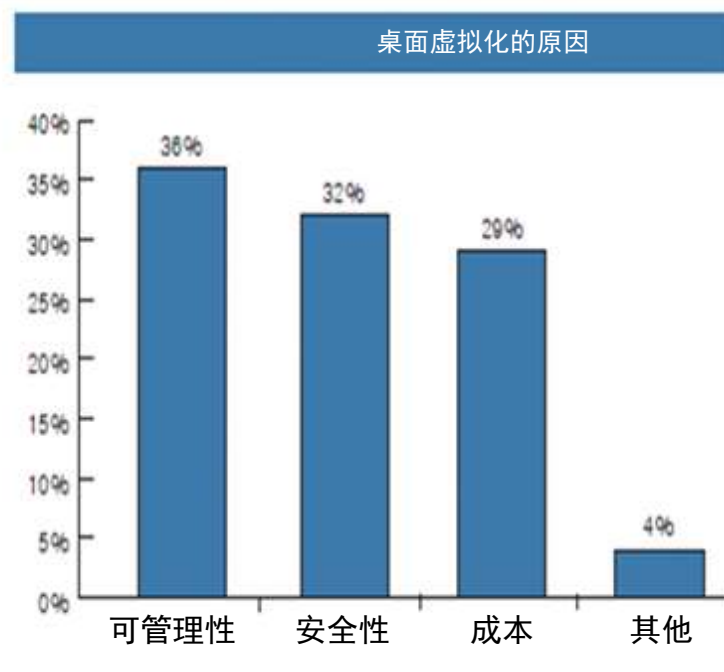


虚拟化可打破各层之间的依赖关系



桌面虚拟化 — 驱动因素

- 可管理性问题
 - ▶ 各种硬件型号、PC 更新换代周期和硬件不兼容性
- 安全性问题
 - ▶ 丢失或被盗的笔记本电脑/台式机
- 成本问题



来源: Goldman Sachs, Independent Insight, US technology Strategy

桌面虚拟化

桌面虚拟化

允许从端点设备分离用户状态、操作系统 (OS) 和应用程序的技术。

- 允许企业承载和集中管理桌面
 - ▶ 桌面在 VDC 中以虚拟机方式运行
 - ▶▶ 它们可以通过局域网/广域网访问
 - ▶ 端点设备可以是精简客户端或 PC
- ▶ 桌面虚拟化技术的目标在于将 PC 的操作系统 (OS) 集中在数据中心。这旨在显著简化桌面的安全性和管理。
- ▶ 数据中心承载的桌面在 VDC 中以虚拟机 (VM) 方式运行，而最终用户从各种端点设备远程访问这些桌面。
- ▶ 端点设备上不会发生应用程序执行和数据存储；所有操作都集中在数据中心完成。



桌面虚拟化的优势

- **启用精简客户端：**桌面虚拟化允许使用精简客户端作为端点设备。这样可以创造机会将陈旧的 PC 替换为精简客户端设备，而精简客户端设备的生命周期通常是标准 PC 的两倍，从而可以显著降低端点硬件的成本。精简客户端与标准 PC 相比，耗电非常低。因此，它们支持企业的“绿色”战略并降低运营成本。
- **提高数据安全性：**由于桌面在企业的数据中心内以虚拟机方式运行，因此降低了数据泄漏和被盜的风险，并简化了法规遵从性过程。
- **简化数据备份：**由于集中式虚拟桌面全部位于企业的数据中心内，因此更容易确保完全符合备份策略。
- **简化 PC 维护：**虚拟桌面的维护比传统的 PC 简单得多。由于虚拟机独一无二的特性，因此可以很方便地修补应用程序、调配/删除用户以及迁移至新的操作系统。
- **访问灵活性：**如果企业的桌面环境使用桌面虚拟化技术集中管理，则可以向不具有企业 PC 访问权限的用户提供桌面环境的访问权限。如果用户需要在家或者离开办公桌办公，或者其他远程办公情形，在这些情况下这一特点尤其有用。



桌面虚拟化技术

- 技术 1：远程桌面服务 (RDS)
- 技术 2：Virtual Desktop Infrastructure (VDI)
- 桌面虚拟化技术允许集中承载和管理桌面环境
 - ▶ 将桌面环境远程提供给用户的端点设备

远程桌面服务

- RDS 通常称为终端服务
- 终端服务在安装 Windows 之后才能运行
 - ▶ 将各个会话提供给客户端系统
 - ▶ 客户端接收桌面的显示
 - ▶ 资源消耗在服务器上发生

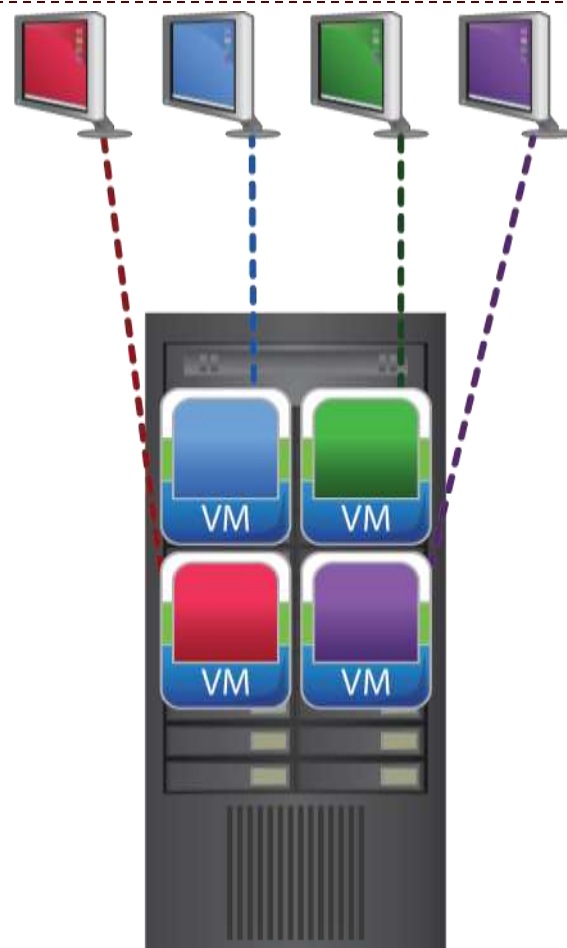


远程桌面服务的优势

- 快速提供应用程序
 - ▶ 应用程序安装在服务器上并从服务器进行访问
- 提高安全性
 - ▶ 应用程序和数据存储在服务器中
- 集中化管理
- 相对于 VDI 的低成本技术

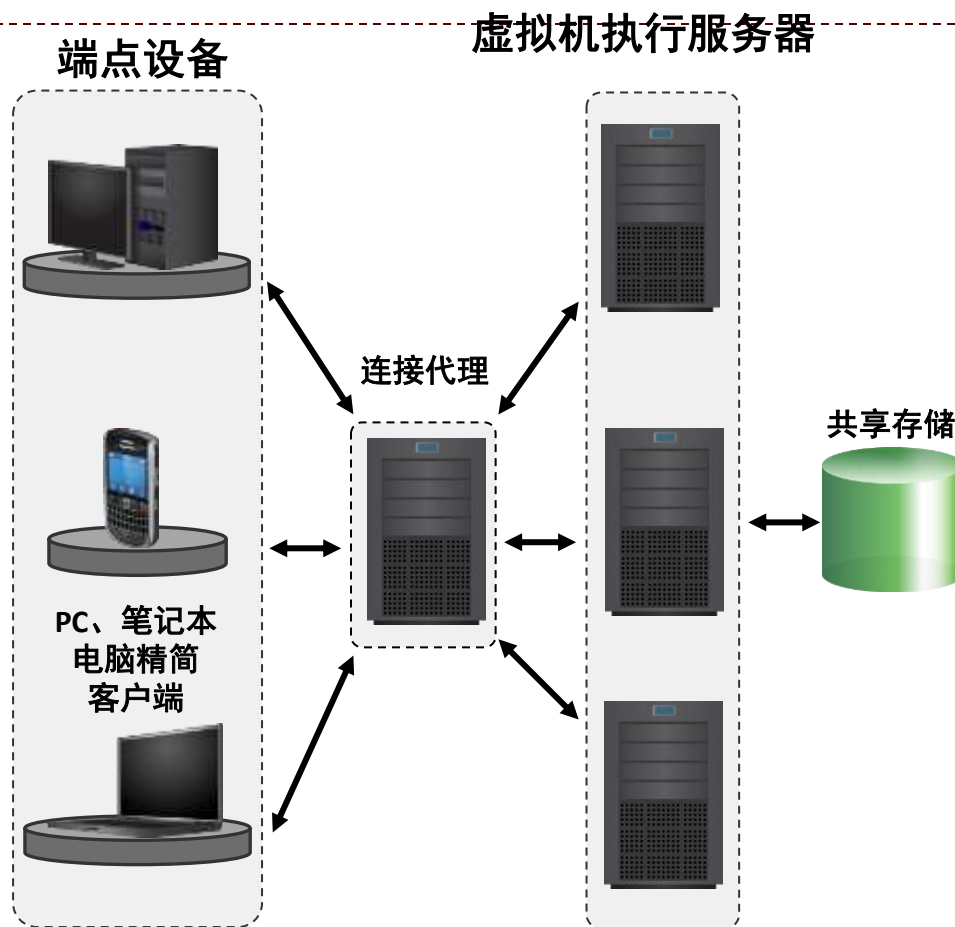
Virtual Desktop Infrastructure (VDI)

- VDI 涉及承载在 VDC 服务器上以虚拟机方式运行的桌面
 - ▶ 每个桌面都安装有自己的操作系统和应用程序
- 用户对虚拟化桌面的资源具有完全访问权限
- 服务器承载的桌面虚拟化解决方案方法有时称为虚拟桌面环境 (VDE)。
- VDI 允许用户通过远程桌面传输协议从端点设备访问远程桌面环境。
- 对于用户，使用体验与使用 RDS 解决方案非常相似，差别是桌面操作系统是运行在服务器承载的虚拟机上，而不是单个服务器操作系统的远程用户会话上。



VDI: 组件

- 端点设备
- 虚拟机承载/执行服务器
- 连接代理

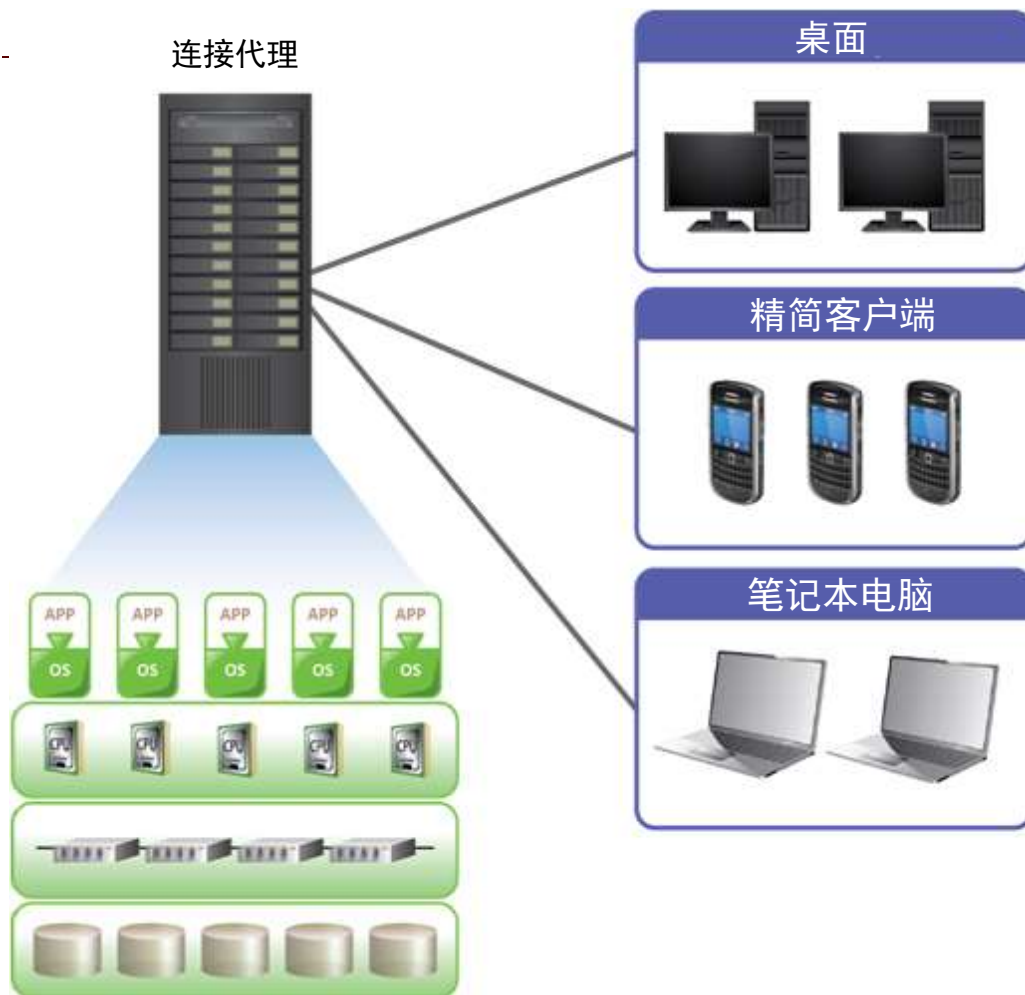


虚拟机承载服务器

- 虚拟机承载服务器负责承载桌面虚拟机
 - ▶ 远程提供给端点设备
- 每个虚拟机都可以专用于特定的用户，或者在池中进行分配
 - ▶ 虚拟机池共享虚拟机以便供多个用户并发使用
- 当对虚拟机进行资源调配时，可以将模板或映像用作虚拟机创建、设置和磁盘的基础

连接代理

- 它负责建立和管理
 - 端点设备与桌面虚拟机之间的连接



VDI：优势和考虑事项

优势

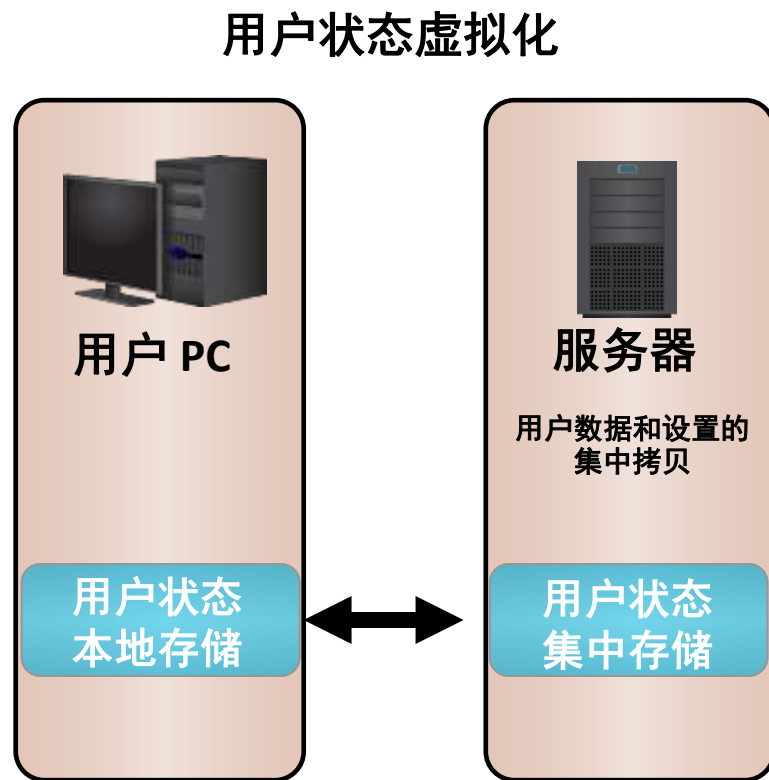
- 集中化部署和管理
- 提高安全性
- 改善业务连续性和灾难恢复

考虑事项

- ▶ 网络连接的依赖性
- ▶ 不适用于高端图形应用程序
- ▶ 需要额外的基础架构

用户状态虚拟化

- 用户状态包括用户的数据以及应用程序和操作系统配置设置
- 用户状态虚拟化允许采用桌面虚拟化的实施技术
- 用户状态虚拟化在中央位置存储用户的数据和设置
- 用户状态虚拟化的优势：
 - ▶ 操作系统更新换代/迁移时更易于迁移用户的状态
 - ▶ 使用户可以在任何端点设备上使用数据



应用程序虚拟化

应用程序虚拟化

它是向最终用户提供应用程序的技术，而无需底层计算平台上的任何安装、集成或依赖关系

- 允许在隔离的环境中提供应用程序
 - ▶ 将操作系统 (OS) 资源和应用程序聚合到虚拟化的容器中
 - ▶ 确保操作系统 (OS) 和应用程序的完整性
 - ▶ 避免不同的应用程序之间或相同应用程序不同的版本之间的冲突

应用程序虚拟化：部署方法

- 应用程序封装
 - ▶ 将应用程序转换为独立的软件包
 - ▶▶ 不依赖于软件安装或底层操作系统
 - ▶ 应用程序软件包可以从 USB、CD-ROM 或本地磁盘运行
 - ▶ 软件包中含有内置代理
- 应用程序流
 - ▶ 执行应用程序时，特定于应用程序的数据/资源将传输到客户端设备
 - ▶ 最低数量的数据（通常为整个应用程序的 10%-30%）传输到客户端
 - ▶▶ 在应用程序启动之前
 - ▶ 根据需要提供其他应用程序功能
 - ▶ 需要本地安装的代理才能运行虚拟化应用程序

应用程序虚拟化：优势

- 简化应用程序部署/淘汰
 - ▶ 不安装应用程序
- 简化操作系统映像管理
 - ▶ 应用程序完全与操作系统分离
 - ▶ 操作系统修补和升级不影响应用程序
- 消除资源冲突
 - ▶ 应用程序具有它们自己的虚拟操作系统资源

谢谢！