



SOFT130091.01

云原生软件技术

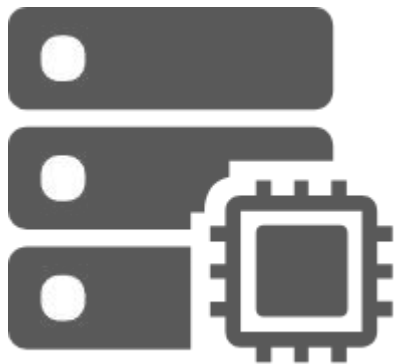
2. 虚拟化与云计算



复旦大学软件学院
沈立炜

shenliwei@fudan.edu.cn

操作系统中常见的资源



计算资源

服务器CPU
内存容量

.....



存储资源

服务器磁盘空间

.....



网络资源

服务器网卡
网络类型、路由器、
网关

.....

直接使用物理设备的不足

• 缺乏灵活性

- ✓ 缺乏时间灵活性：当系统中资源匮乏，需要扩容资源时，物理设备的供给流程很长，设备到货后安装、部署设置烦琐，从而导致时间灵活性差。
- ✓ 缺乏空间灵活性：用户可能需要很小规格的计算资源，为此购置一台服务器会造成资金的浪费；为了应对计算资源的峰值，购置了性能足够的资源，然而大多数时候资源开销远远低于峰值，导致资源利用率低下。
- ✓ 缺乏操作灵活性：直接使用物理设备时，设备上架、安装、配置等都需要人工完成大量的工作，或者针对网络配置，需要对许多不同的网络设备进行统一管理。

直接使用物理设备的不足

- 隔离性差

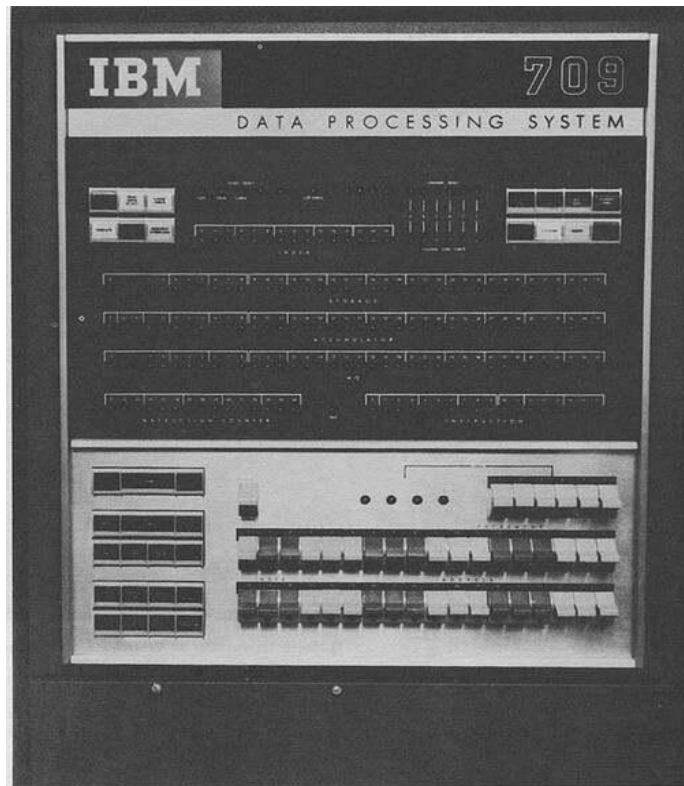
- ✓ 为了提高资源利用率，可能会将多个服务部署到同一个物理设备上，这可能会导致多个服务之间相互干扰、隔离性差，严重时甚至会造成安全问题。

- 不可复制

- ✓ 当一个物理设备上的应用需要扩展时，只能在增加物理设备后，再做一遍物理资源配置、应用发布、网络配置等工作，无法快捷地实现对应用的复制和粘贴。

虚拟化原理

- 虚拟化技术最早出现于1961年，IBM 709计算机首次将CPU占用切分为多个极短的时间片（ $1/100s$ ），每一个时间片都用来执行不同的任务。
- 通过对这些时间片的轮询，就可以将一个CPU虚拟化或者伪装成多个CPU，并且让这些虚拟CPU看起来同时在运行。
- 简单来说，虚拟化就是通过模仿下层原有的功能模块创造接口来“欺骗”上层的机制。



虚拟化原理

- 虚拟化可以通过资源管理技术，将物理存在的实体资源以虚拟机（Virtual Machine, VM）的形式抽象成一种逻辑表示。
- 通常的做法是在整个系统架构中增加一个抽象层，负责分割下层的物理资源，然后组合成逻辑资源供上层使用。

广义虚拟化

- 硬件虚拟化

- ✓ 硬件抽象层虚拟化为上层虚拟一个相同的或者相似的硬件

- 内存虚拟化

- ✓ 虚拟内存是对整个存储空间的虚拟化，使得应用程序可以突破物理存储的限制

- 系统虚拟化

- ✓ 操作系统层虚拟化为上层模拟一个完备的操作系统，实现对应用进行独立封装，提供隔离的环境

- 应用虚拟化

- ✓ 在应用层及函数层之上建立抽象层，提供逻辑隔离功能，即允许在一个应用上同时支撑多个用户应用（多租户）

狭义虚拟化

- 狭义虚拟化特指操作系统虚拟化，它使用虚拟化技术在一台物理机上模拟出多台虚拟机
- 每台虚拟机都拥有独立的硬件设备（CPU、内存、硬盘、网卡等），上面运行着一个独立的操作系统，通常称之为“GuestOS”，每个 GuestOS 中都运行着各自的应用程序
- 虚拟化的资源以虚拟机的形式提供给用户，用户在使用虚拟机时可以选定不同的资源规格，包括CPU、内存和存储资源等，同时虚拟机可以挂载不同的存储，被配置在不同的网络中

狭义虚拟化

- 虚拟化的定义：
虚拟化为应用程序的运行提供了包含软、硬件资源的仿真环境。这一组仿真环境被称为虚拟机。

美国国家标准与技术研究院

NIST

**National Institute of
Standards and Technology**

U.S. Department of Commerce

Baidu 百度

狭义虚拟化

- 虚拟化可以提供与底层硬件设备系统不同的虚拟机
- 虚拟化可以虚拟出多台虚拟机资源
- 虚拟化的具体实现是通过在操作系统与硬件之间加入一个虚拟化层，并通过空间上的分割、时间上的分时及仿真模拟，将服务器物理资源抽象成逻辑资源
 - ✓ 虚拟化层可以将单个CPU模拟为多个CPU，并且这些CPU之间相互独立、互不影响，也就是虚拟化层实现了计算单元的模拟及隔离
 - ✓ 向上层操作系统提供与原先物理服务器一致的环境，使得上层操作系统可以直接运行在虚拟环境中，并允许具有不同操作系统的多台虚拟机相互隔离，并发地运行在同一台物理服务器上

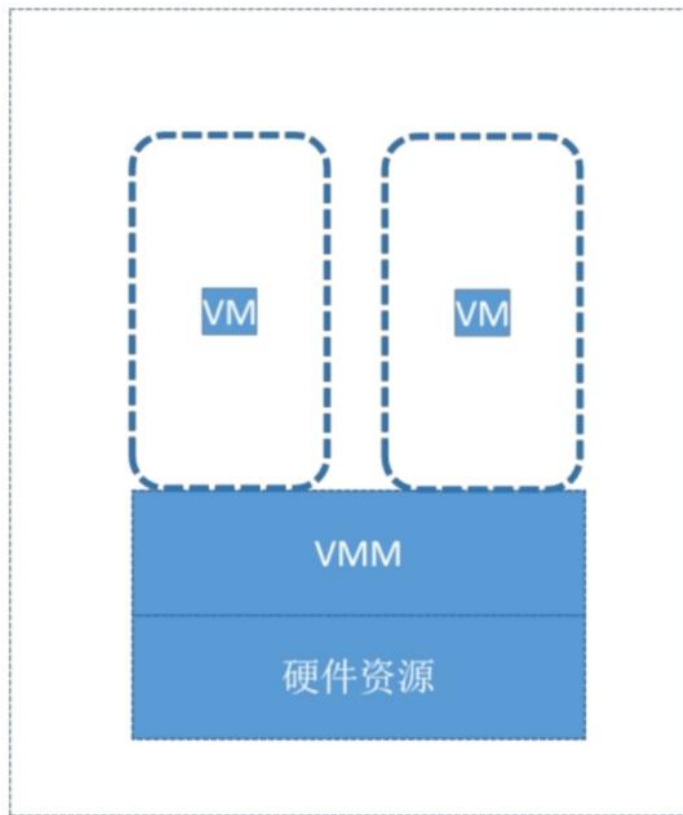
狭义虚拟化

- 虚拟化场景

- ✓ 硬件资源
- ✓ 虚拟机监视器 (VMM)
- ✓ 虚拟机 (VM)

- 虚拟机的特征

- ✓ 分区
- ✓ 隔离
- ✓ 封装
- ✓ 独立



常见虚拟化产品



VirtualBox



虚拟化类型

- 全虚拟化
- 半虚拟化
- 硬件辅助虚拟化

全虚拟化

- 全虚拟化是一种虚拟化技术，通过在硬件和操作系统之间添加一个虚拟机监控器（VMM）或Hypervisor，完全模拟底层硬件环境，使虚拟机可以运行任何操作系统（GuestOS）而无需修改。
- 面临性能开销较大、资源消耗较多、配置复杂等问题。



半虚拟化

- 全虚拟化是一种虚拟化技术是在全虚拟化的基础上发展起来的，通过修改客户操作系统（Guest OS）的内核，使其能够直接与Hypervisor交互，从而减少性能开销。
- 具有需要修改客户操作系统、兼容性受限的不足。



硬件辅助虚拟化

- 全虚拟化通过软件模拟硬件环境，为虚拟机提供完整的虚拟化资源。
- 硬件辅助虚拟化则利用硬件提供的虚拟化扩展技术来加速虚拟化过程，提高虚拟机的性能和效率。



虚拟化架构

- 裸金属架构 Type1
- 宿主模式架构 Type2

裸金属架构

- 虚拟机监视器被直接安装和运行在物理机上，依赖其自带的虚拟内核管理，使用底层硬件资源
- 虚拟机监视器拥有硬件的驱动程序，不依赖特定的操作系统，其管理着宿主机及其他虚拟机
- 宿主机和虚拟机都安装有各自的操作系统，即宿主机操作系统和GuestOS

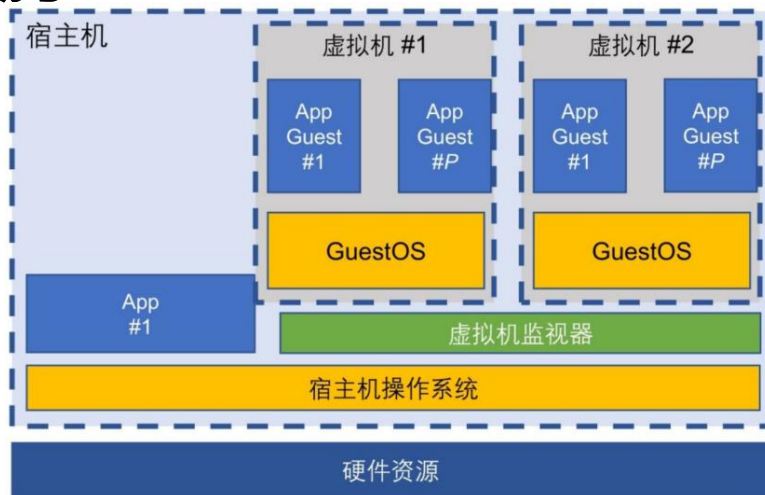


宿主模式架构

- 虚拟机监视器被安装和运行在操作系统上，依赖操作系统对硬件设备的支持和对物理资源的管理
- 虽然虚拟机监视器对硬件资源进行访问必须经过宿主机操作系统，但虚拟机监视器依然可以充分利用操作系统对硬件设备的支持，以及内存管理进程调度等服务



VirtualBox



虚拟化的不足

- 虚拟化软件解决了硬件资源利用率的问题
- 虚拟化软件可以用于创建虚拟机，但需要事先人工指定将虚拟机放在哪台物理服务器上
- 虚拟化技术缺乏以下三大灵活性
 - ✓ 时间灵活性
 - ✓ 空间灵活性
 - ✓ 操作灵活性

虚拟化的不足

- 缺乏时间灵活性

- ✓ 虚拟化服务并非随时可用，服务商无法随时确保服务的可用性。同时虚拟化的过程还需要比较复杂的人工配置，随着集群规模的扩大，人工配置愈加复杂、耗时。所以，仅通过虚拟化软件管理的物理服务器规模并不是特别大，一般是几十台至百台的规模

虚拟化的不足

- 缺乏空间灵活性

- ✓ 缺乏分布式部署，无法实现资源弹性扩展。
当用户数量增多时，虚拟化软件所能管理的集群规模远未达到理想的程度，很可能造成资源不够。所以，随着对集群规模的需求越来越大，必须采取自动化的流程来实现资源弹性扩展

虚拟化的不足

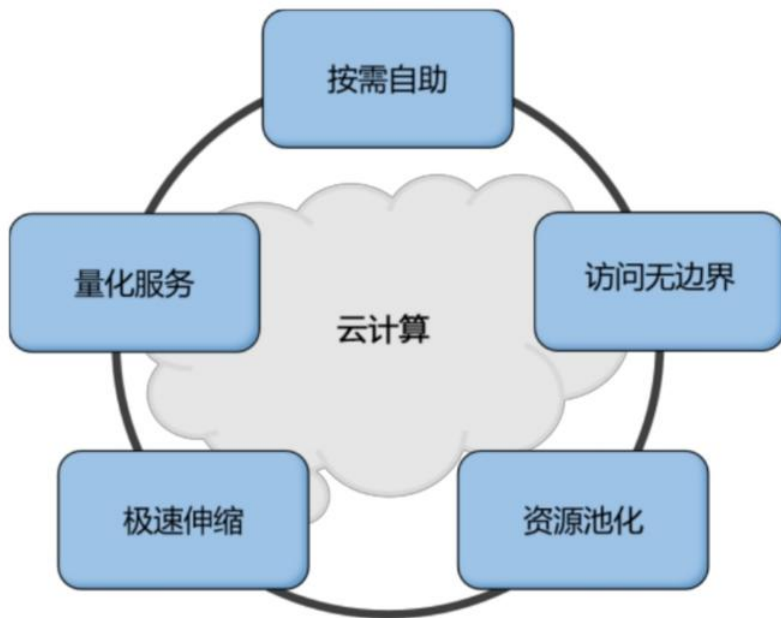
- 缺乏操作灵活性
 - ✓ 缺乏统一的自动化管理
 - ✓ 虽然创建一台虚拟机的过程相对较为基础，但是操作的灵活性意味着用户可以更方便、迅速地对资源进行更进一步的管理操作，所有操作都可以通过自动化脚本来实现

云计算的目标

- 云计算的目标是解决这三大灵活性的问题
- 云控制平台通过调度器（Scheduler）管理由几千台物理服务器抽象而来的虚拟资源池，无论用户需要多少CPU、内存、硬盘资源的虚拟机，调度器都会自动在资源池中匹配到最合适的资源，通过虚拟机的形式供给资源，并做好配置
- 这个阶段称为“池化”或者“云化”
- 虚拟化的能力和灵活性到了这个阶段，才可以被称为“云计算”，在这之前都只能叫作“虚拟化”

云计算的特点

- 云计算指IT基础设施/服务的交付和使用模式，用户可以通过网络，以按需、易扩展的方式获得所需资源/服务



云计算的特点

- 按需自助
 - ✓ 用户无须与服务提供商交互，就可以自动地得到自助的计算资源能力
- 访问无边界
 - ✓ 借助不同的客户端，通过标准的应用对网络访问的可用能力
- 资源池化
 - ✓ 根据用户的需求动态地划分或释放不同的物理资源和虚拟资源，这些池化的计算资源以多租户的模式来提供服务

云计算的特点

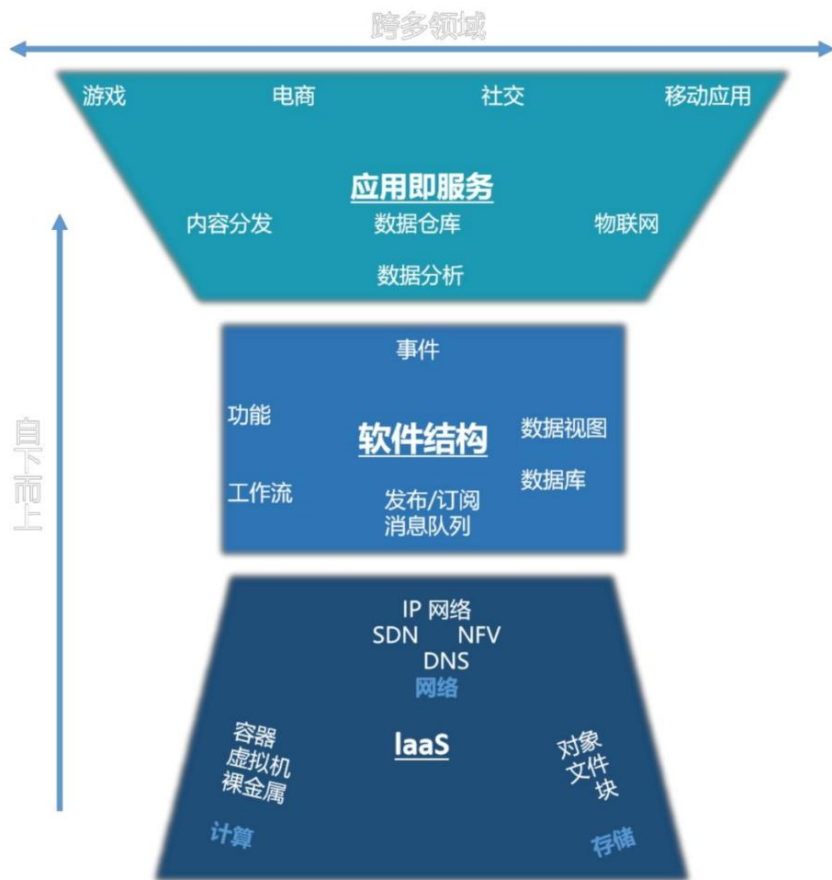
- 极速伸缩
 - 一种对资源快速和弹性提供的能力以及释放的能力。对用户来讲，这种能力是无限的（随需的、大规模的计算机资源），并且可在任何时间以任何量化方式购买
- 量化服务
 - 云系统通过计量的方法来自动控制服务的类型，优化资源的使用，例如存储、带宽以及活动用户数。监测和控制资源的使用，为供应商和用户透明报告（即付即用的模式）

云计算的本质

- 云连接着网络的另一端，为用户提供了可以按需获取的弹性资源和架构。用户按需付费，从云上获得所需的计算资源，包括存储、数据库、服务器、应用软件及网络等资源，这种从资源到架构的全面弹性，降低了运营成本，更加契合变化的业务需求
- 云计算让计算资源像用水和用电一样方便，从而大大提高计算资源的利用率和用户的工作效率
- 云计算模式可以被简单地理解为，不论是服务的类型，还是执行服务的信息架构，依托互联网向用户提供应用服务，使其不需要了解服务器在哪里、内部如何运作，通过浏览器即可使用

云计算服务模式

- 基础设施即服务 (IaaS)
- 平台即服务 (PaaS)
- 软件即服务 (SaaS)



IaaS：基础设施即服务

- 计算、网络、存储等资源常被称为“基础设施（Infrastructure）”，管理这些资源的云平台往往被称为“基础设施服务平台”（IaaS）
- IaaS实现了时间灵活性、空间灵活性和操作灵活性，通过调度器动态地管理计算、网络、存储等资源

云的部署模式

	私有云	社区云	公有云	混合云
云基础设施用户	一个单一组织	有共同愿景的组织	对公众公开	混合云是私有云、社区云和公有云的有机结合： <ul style="list-style-type: none">• 保持独立实体• 通过各自的机制整合• 让数据和应用的使用更便捷
云基础设施拥有者	机构、第三方、联盟等	机构、第三方、联盟等	企业、协会、政府组织、联盟等	
云基础设施归属	外购 内部	外购 内部	云平台供应商	

- 私有云：将虚拟化软件和云控制平台等部在用户的数据中心
- 社区云：将虚拟化软件和云控制平台等部署在有共同愿景的组织内部或外部
- 公有云：将虚拟化软件和云控制平台等部署在云厂商的数据中心，用户不需要很大的投入，只要注册一个账号，就能通过Web服务访问、管理云资源
- 混合云：通过对私有云和公有云的统一管理，按需将不同安全等级的应用部署到不同的云上。同时，混合云还提供了异地容灾能力，将主要业务部署在私有云上，将备用业务部署在公有云上，降低了成本，而且灾难恢复被控制在很短时间内

IaaS的主要功能

- 资源接入与抽象
 - ✓ 通过虚拟化及软件可定义方式，将底层虚拟化资源抽象为可识别的计算、存储、网络等资源池，以此作为云控制器对各类硬件资源实施管理的基础
- 资源分配与调度
 - ✓ 利用云控制器的资源管理能力，按照不同租户对资源类型和数量的不同需求，将资源分配给各个租户
- 应用生命周期管理
 - ✓ 协助租户实现虚拟机在云上的安装、启动、停止、卸载等管理操作
- 云平台管理维护
 - ✓ 通过云平台管理员实施对整个云平台的各类管理及运维操作
- 人机交互支持
 - ✓ 提供人机交互界面，支持云平台管理员及普通租户对云平台实施各类操作

IaaS架构

安全层

物理安全

主机安全

数据安全

网络安全

应用安全

业务安全

边界安全

专家服务

云平台安全

认证审计

云管理层

多云异构纳管

混合云

云服务控制层

资源池管理

多租户隔离

弹性伸缩

工单处理

软件定义
存储

软件定义
网络

虚拟化资源层

计算

虚拟机服务

存储

块存储

对象存储

文件存储

网络

私有网络

负载均衡

专线/VPN接入

虚拟网卡

物理资源层

计算设备

x86
服务器

ARM
服务器

GPU
服务器

存储设备

分布式存
储服务器

FC-SAN存
储服务器

网络虚拟化

交换机
Switch

路由器
Router

物理防火
墙

运维层

监控告警

自动化部署

故障追查

操作日志

报表体系

知识库

服务与支持

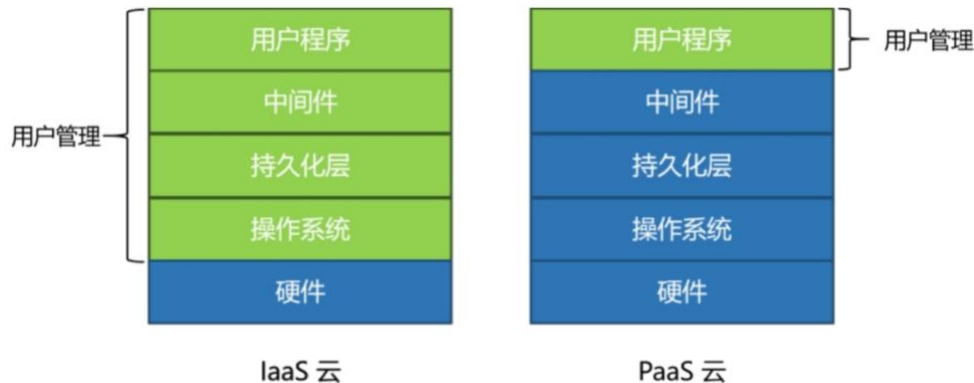
工单处理

大屏展示

流程管理

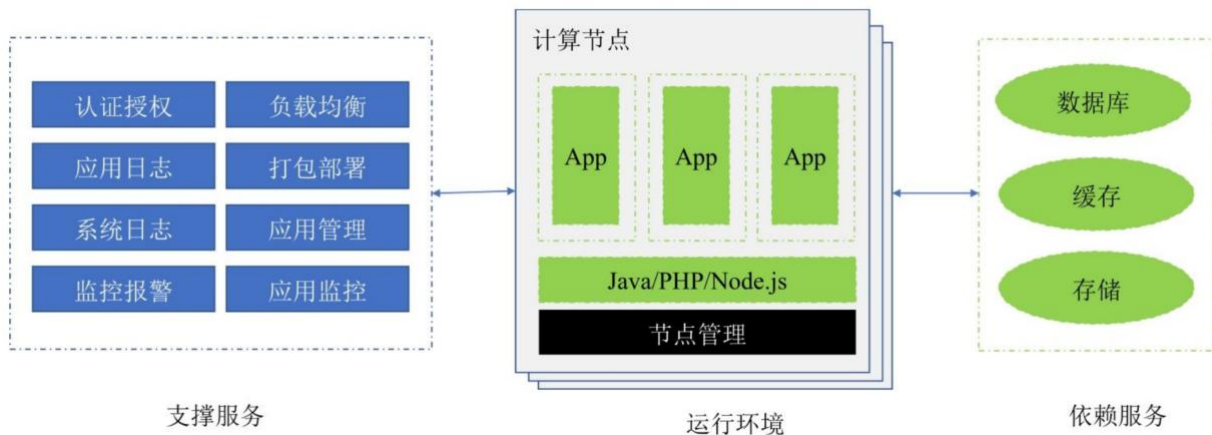
PaaS：平台即服务

- PaaS不但基于IaaS提供了虚拟化资源，同时还基于虚拟化资源自动化部署应用运行时所依赖的库、工具、服务、运行时环境等
- PaaS的核心理念是用户只需要在 PaaS 之上部署应用，PaaS 可以确保底层的稳定性，从而使用户把更多精力投放在自己的应用中
 - 在使用IaaS服务时，用户仍需要对操作系统、中间件、数据库等进行日常维护，这不仅加大了使用的复杂度，而且还大大增加了日常维护的工作量
 - 在PaaS服务模式下，用户无须关心虚拟机、操作系统、存储，可以自由地在虚拟机上安装所需的应用



PaaS的理想工作模式

- 应用开发者提交应用代码，PaaS平台的代码托管工具实现代码的编译、打包。打包后的代码通过PaaS平台的自动化部署模块在IaaS提供的虚拟机上自动部署，此部署过程涉及整个代码运行环境的设置。一旦代码部署完毕，PaaS平台就会为该应用自动提供所有依赖的外部服务，从而确保应用正常运行。在应用运行过程中，PaaS平台还会自动接入日志、监控、告警等非核心业务服务，保障应用的整个运行过程



PaaS的核心功能

- 自动化部署
- 多副本管理
- 弹性伸缩

自动化部署

- PaaS提供了自动化编译、打包、部署的核心能力
- 基于已有的IaaS，先前主流做法是租一批虚拟机，然后像使用物理服务器一样通过脚本或者手工方式在这些虚拟机上部署应用。在部署过程中难免会遇到云端虚拟机和本地环境不一致的问题，而PaaS的核心目的就是解决这个问题
- Cloud Foundry 是 VMware 在 2011 年开源推出的，它通过对应用的直接管理、编排和调度，让开发者专注于业务逻辑而非基础设施。在 Cloud Foundry 中，当虚拟机创建好之后，只需要在虚拟机上部署 Cloud Foundry，然后开发者执行一条命令就能把本地应用部署到虚拟机上

多副本管理

- 为了保障PaaS服务的高可用性，需要通过多副本机制实现应用的高可用性
- 多副本机制将应用及相关数据复制到不同的节点上，从而确保即使有部分节点损坏，也不会影响整体的PaaS服务
- 因为相同的数据有多个副本同时存在，所以在数据源更新之后需要进行副本数据的同步
- 同步方式有推（Push）和拉（Pull）两种
 - ✓ 推是指从数据源向各个副本推送更新数据
 - ✓ 拉是指各个副本根据同步算法从数据源拉取更新数据

弹性伸缩

- 虽然多副本机制可以保证PaaS服务的高可用性，但是无法保证PaaS有足够能力服务于海量的用户
- PaaS 通过垂直或水平的弹性伸缩来实现高可用性
 - ✓ 垂直弹性伸缩是指通过在单节点上添加CPU、内存等资源实现伸缩
 - ✓ 水平弹性伸缩是指通过添加节点实现伸缩，目前更多地采用水平弹性伸缩
- 在对应用进行弹性伸缩时，需要考虑以下两点
 - ✓ 应用无状态化
 - ✓ 有状态应用分区

PaaS的优缺点

- 优点

- ✓ 灵活性：应用自动化部署
- ✓ 高可用性：通过多副本实现
- ✓ 高性能：通过弹性伸缩以及自动化部署实现
- ✓ 隔离性：多租户逻辑隔离或物理隔离

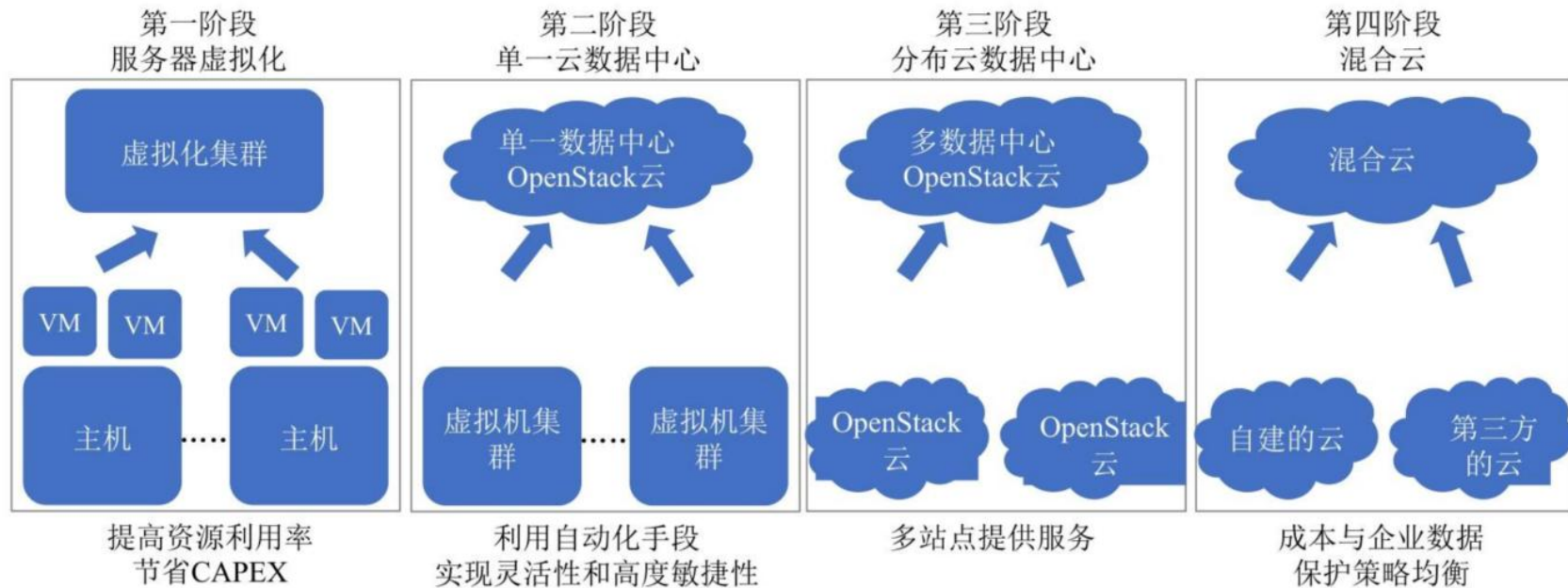
- 缺点

- ✓ 传统PaaS使得应用与PaaS平台之间有着非常强的耦合性，PaaS为应用提供了专属的SDK，应用必须依赖这些SDK，而用户必须使用PaaS平台提供的框架和中间件来重新开发自己的应用

SaaS：软件即服务

- SaaS以服务的形式对外提供应用程序的能力，其本质等同于通过SOA形式对外暴露服务，供最终用户使用
- 虽然在理论讲述或讨论时，SaaS分层总在PaaS之上，但是大部分SaaS不是基于PaaS开发的，而是直接在IaaS上开发的。所以可以说，SaaS和PaaS是两种不同的云计算对外提供服务的方式

云计算架构的演进过程



云计算的发展

- 云计算1.0：IT 基础设施虚拟化
- 云计算2.0：IT基础设施IaaS化
- 云计算3.0：多云级联
- 云计算4.0：应用的云原生化
 - 除了云平台本身技术的进步，云上的应用也需要逐步从单体、小规模、有状态、进程式、烟囱式向云原生应用模式转变
 - 云原生化主要是通过容器编排技术来实现应用的分布式和高并发要求，保障应用的快速启动、大规模管理、动态编排，从而实现应用自身的上云改造

云计算的发展大事件

- 2006年：Amazon率先推出了基于开源Xen开发的EC2弹性伸缩服务，使用户可以在公有云上申请应用所需的虚拟机
- 2008年：Google发布了Google App Engine，是一款让用户可以在Google的IaaS上运行应用程序的平台产品
- 2009年：Heroku推出了第一款PaaS服务
- 2010年：Microsoft推出了IaaS+PaaS平台Azure；Rackspace和NASA正式发布了开源的IaaS云控制软件OpenStack，帮助云服务商和企业构建类似于Amazon EC2和S3的IaaS，目前已成为IaaS私有云的事实标准

云计算的发展大事件

- 2011年：发布了开源的PaaS平台Cloud Foundry
- 2013年：Google正式推出了IaaS服务Google Cloud Engine，宣告云计算三大巨头Amazon、Microsoft、Google的大战开始
- 2014年：Amazon推出了首款FaaS（Function as a Service）产品Lambda，标志着云上应用进入Serverless（无服务）时代
- 2015年后，云计算本身的发展进入了比较平缓的阶段，整个行业注意力转移到了更为热点的“云原生”上

内容小结

- 虚拟化的原理
- 虚拟化类型
- 虚拟化架构
- 云计算的目标、特点与本质
- 云计算服务模式
- IaaS
- PaaS
- SaaS

SOFT130091.01

云原生软件技术

End

2. 虚拟化与云计算