

虚拟化与容器技术

大家好，我是胡才郁，我介绍的主题呢，是虚拟化与容器技术。如今是大数据和云计算的时代，而这类应用高速的发展呢，离不开虚拟化与容器化。

我是这个学期第一个演讲的同学，我希望今天呢，讲的内容有趣一点，不至于因为上网课大家都不愿意听，并且我认为今天将要讲到的docker技术对于我们学生而言是十分有用的，所以讲到那个部分的时候呢，也会有动手实操的环节，希望不会那么枯燥。

封面上的logo呢，左边的是Vmware，是虚拟化技术的代表，右边的是docker，是容器技术的代表。要理解虚拟化与容器化技术，我们首先要明白的是，为什么我们需要虚拟化与容器化技术。[翻页](#)

落后的旧时代

来看一个例子。这张图大家肯定再熟悉不过了，在秋季学期考试出成绩的时候，大家都在学校服务器上都挤着登陆，就造成了这样的情况发生。这张图上显示的用户名或密码错误其实并不是大家登录不上去的真正原因，你会发现过了查询成绩的高峰期就可以登陆上去了。实际上这是由于访问教务处的流量太多了，把学校服务器给弄崩了。



在大数据和云计算的背景下，我们当今的互联网服务具有**数据量大**这样的特点，而这样的特点直接造成的影响就是，我们对于运行着业务的机器，性能要求是很高的。像这样的场景还有很多很多，比如说前两天的双十二购物节，我们能够成功在零点秒杀商品，也依赖于阿里云的高性能服务器集群。[点击](#)

在过去的时代，每个服务器只能运行单一应用，这并不是我们不想运行多个，主要原因是如果同时运行多个应用程序的话，它们之间可能会产生干扰，影响系统的性能与稳定性。并且如果其中一个应用程序压力过大挂掉了，其它应用程序也可能遭受影响，导致整个系统的安全性降低。

所以在过去，如果我们想要新增一个应用，就意味着要去买一台新的服务器。但是大部分情况下，我们是不清楚买配置多高的机器就带得动这个新应用。但是为了这个新应用可以稳定的运行，买服务器时难免买到性能大幅度高于实际需求的服务器。毕竟大家都不希望会因为查成绩的同学太多了，并发太高，就把学校服务器搞崩了。

那基于这样的背景，诞生了一项具有跨时代意义的技术，就是**虚拟化技术**。

[翻页](#)

虚拟化技术

虚拟化技术的核心内涵就是可以在**一台**物理计算机上创建出**多个**虚拟计算机，在这种情况下，每个虚拟机都有自己独立的**操作系统、硬件设备和软件资源**，他们共享着同一台物理机。

虚拟化技术做的最好的公司就是VMware，他的公司logo在这里，（鼠标指）。他是第一个将虚拟化技术商用的公司。很多同学使用的虚拟机软件就是用他们公司开发的Vmware station。对于我们而言，虚拟机的一个很常见的应用场景就是，我们在做操作系统实验时，在自己的windows操作系统上虚拟化出Linux操作系统。

相比较于过去的单机单应用的旧时代，虚拟化技术有着独特的优势。

- 首先他实现了资源共享，多台虚拟机共享一台硬件资源，大幅度的提高了系统的利用率。
- 并且它减少了部署和管理的工作量；虚拟化技术对应用程序有效隔离，避免一个应用程序的安全漏洞影响到其它应用程序。

但是这么一项伟大的技术也没有做到完美，否则今天演讲的标题可能就要划掉容器技术了。我们来思考一个问题，（点击），虚拟化技术是做到了硬件层面的隔离，那操作系统层面呢。

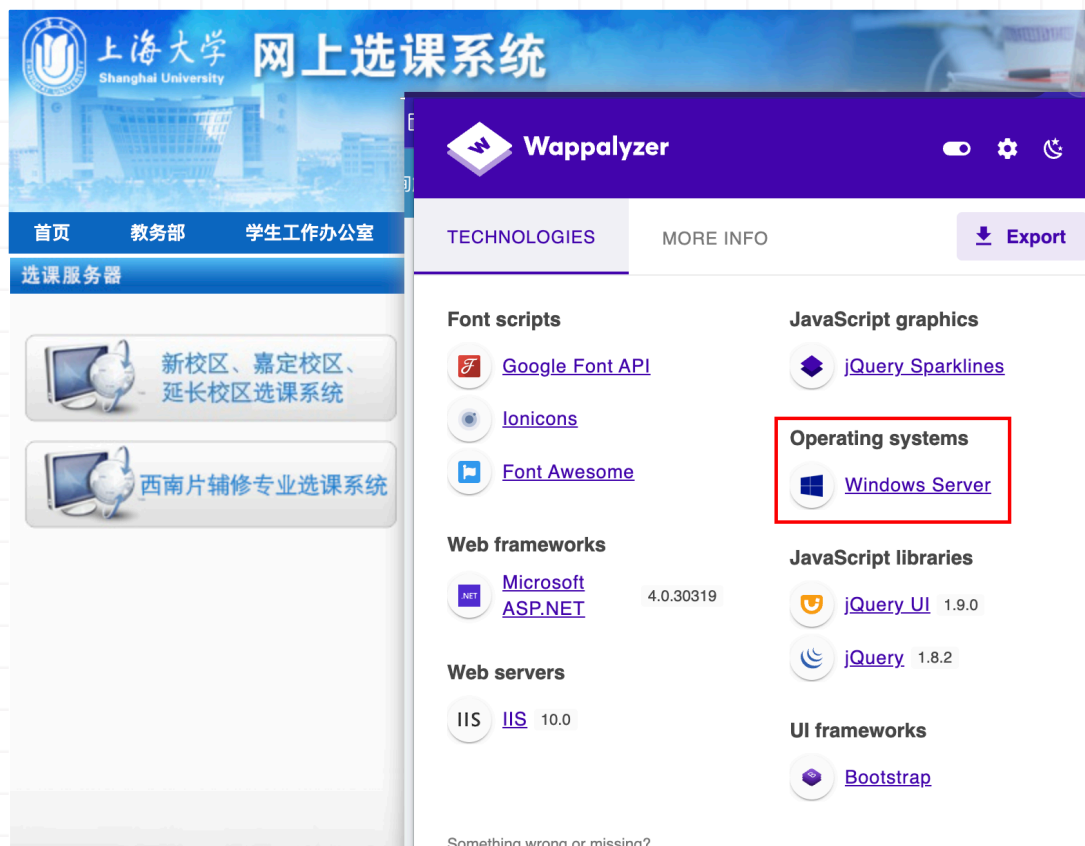
硬件的成本的确节约了，例如对于某一个应用，我想分配给他4G的内存，64G的存储，或者更大更小，这些都是可以由我们自己来动态调整的，但是对于操作系统怎么办呢？

其实，这也是虚拟机技术最大的缺点，他依赖于专用的操作系统。我们所希望的最简单粗暴的想法，就是只要我分配了这4个G内存，那么这4个G的资源就可以完全交给我们的应用来使用。但是在实际中，这是不可能的，我们需要操作系统来帮助管理我们的应用，但是操作系统会占用额外的CPU、内存和存储，这些硬件资源实际上也是一种浪费，只不过是说浪费在了操作系统上。

这只是一方面，另一方面，操作系统也是需要钱的！

（点击）

继续回到之前选课的那个例子上，上边这张图是我用一个插件，查到了咱们学校的服务器用的是Windows Server运行着像选课这样的应用，但是我们知道，微软它不开源，Windows Server可是要钱的！



下面那张图是我在Window官网查找到的价格，Windows Server的基础版也要500刀，这个价格也不便宜，我没有具体看这个许可证可以用于几台机器，从这个角度想想，如果解决了依赖于操作系统这一个缺点，学校肯定就有更多的资金用于提升我们访问网站的体验了，不过前提是信息办的老师们有这个想法。[翻页](#)

Pricing and licensing overview

The following page is intended to provide you with reference pricing for Windows Server 2022. For specific pricing, please contact your Microsoft reseller.

Windows Server 2022 Edition	Ideal for	Licensing model	CAL requirements ^[1]	Suggested Retail Price (MSRP) ^[4]
Datacenter ^[2]	Highly virtualized datacenters and cloud environments	Core-based	Windows Server CAL	\$6,155
Standard ^[2]	Physical or minimally virtualized environments	Core-based	Windows Server CAL	\$1069
Essentials	Small businesses with up to 25 users and 50 devices	Specialty servers (server license) ^[3]	No CAL required	\$501

容器化技术

基于这样一个痛点，容器技术应运而生。容器技术其实类似于虚拟化技术，也是对硬件资源做了虚拟化，但是他很好的解决了虚拟化技术会[独占操作系统](#)这一缺点。

Docker介绍

提到容器，不得不说的就是[Docker](#)。Docker是一个开源的、轻量级容器引擎，用于创建、管理和编排容器。“Docker”一词来自英国口语，意为码头工人。它的logo进行过变更，PPT上边的是docker的老版本的logo，看起来也能很直观的知道它是干嘛的，这个鲸鱼的背上背着的全都是集装箱，也就是容器嘛。和 VMware 虚拟机相比，Docker使用容器承载应用程序，而不使用操作系统，所以它的开销很少，性能很高。[翻页](#)

虚拟化VS容器化

我们来直观的对比一下虚拟化技术和容器化技术在架构上的差异，下面的图展示了有4个业务分别跑在这两种架构上。此处的物理机器可以是笔记本，也可以是云服务厂商像阿里云腾讯云之类的云服务器。

虚拟化实现

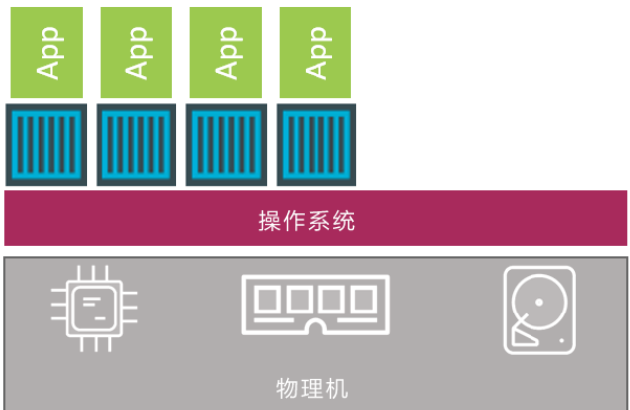
先来看虚拟机模型，在虚拟机模型中，首先要开启物理机并启动Hypervisor引导程序。这个Hypervisor是创建和运行VM的软件。一旦Hypervisor 启动，就会占有机器上的CPU、内存等等物理资源。hypervisor允许一台主机共享物理机的硬件资源，来支持多个跑在上面的多个VM。

所以如果使用虚拟机架构来运行这4个应用的话，在Hypervisor之上需要创建4个虚拟机，并安装4个操作系统，之后再在这4个操作系统上分别跑这4个应用。



容器实现

而对于容器架构呢，与虚拟机模型相同，操作系统也占用了全部硬件资源，不过这里的操作系统只有一个，就是物理机的操作系统。在操作系统层之上，安装Docker的容器引擎。容器引擎可以获取系统资源，比如进程树、文件系统等等，接着将资源分割为安全的互相隔离的资源结构，也就是docker容器。每个容器看起来就像一个真实的操作系统，这里4个容器上分别运行这4个应用。



所以虚拟化与容器化之间的对比就十分直观

- 虚拟化技术是将硬件资源划分为虚拟资源。
- 而容器是将操作系统资源划分为虚拟资源。

所一为什么说容器技术比虚拟机技术更加优秀呢，原因就是容器技术把操作系统也一起虚拟化了。

这样做的好出在于，**操作系统本身是有其额外开销的**。正如我们之前介绍是看到，教务处网站的Windows Server许可证购买是需要钱的，并且操作系统会占用原本属于应用的物理资源。而容器模型只跑一个操作系统，它只有一份操作系统损耗。[\(翻页\)](#)

另一个值得考虑的事情是启动时间。因为容器并不是完整的操作系统，所以其启动要远比虚拟机快。在容器内部并不需要操作系统内核，那么在原先在内核启动过程中对硬件的遍历和初始化这个过程就被跳过了，容器可以在1s之内启动，大家可以想想自己的电脑开机要多长时间。

这就是容器模型要比虚拟机模型简洁，并且高效的原因了。使用容器可以在更少的资源上运行更多的应用，启动更快，并且更便宜，同时面对未知攻击的风险也更小。

Docker扩展

到此，关于虚拟化技术与容器技术的优缺点和原理基本就介绍完了。我们暂时可能不会去当运维人员或者去考虑亲自在服务器部署应用这类事情，所以之前讲的这些呢，对于我们而言其实距离还是比较远的，但接下来的部分我个人感觉是特别实用的，Docker无论是读研还是去工作，绝对是应当作为常识的一门技术。接下来我从一个学生实用的角度进行介绍。[翻页](#)

docker全平台支持良好

首先，docker的跨平台支持性很好，无论你是Windows、Mac还是Linux，都可以安装下载，并且如果你的机器是mac的话，它还对m1芯片做了优化，这张蓝色的图是docker官网的下载页面。并且呢，各个主流IDE平台都提供了插件支持，像Vscode或者jetbrains全家桶里都是直接捆绑安装的，像这个图中写的就是bundled意思就是说安装IDE的时候docker的插件也就一起打包安装好了。[\(翻页\)](#)

一键配置环境

很多同学都接触过深度学习，包括这学期的研讨选题里也有很多这方面的内容。深度学习实践离不开的就是亲自动手训练模型，训练的第一步就是为各种框架配置环境。当自己上手的时候我们往往会遇到大量的坑，各种环境冲突，版本不适配的报错就接踵而至。像左边这张图是关于pytorch安装的版本选择，包括Python包管理器是选择pip还是conda，cuda驱动版本是多少，是不是和自己的显卡适配？这些问题第一次接触的小白很难说清楚，但是如果你用docker部署pytorch的话，一行docker命令就可以解决这个问题。说这个话题我觉得还是很实际的，因为我看有同学的研讨选题会讲到pytorh、tensorflow，包括华为的mindspore，可能她到时候就会遇到这些问题。[\(翻页\)](#)

包括许多很多著名的模型在docker hub上都有自己的镜像，像目标检测领域一个重要的模型yolo v5，在docker中一行命令就可以安装。还有右边下节课就会讲到的Redis数据库等等。docker很好的解决了配环境太复杂这个痛点。我们可以直接使用别人的镜像，用镜像创建出容器之后，就进入了别人搭建好的环境，我们只需要提供硬件支持就可以了。[\(翻页\)](#)

docker容器与镜像

在Docker中，最重要的两个概念就是镜像与容器，我们可以从两个方面直观的理解这两个概念。

- 如果用虚拟机类比的话，Docker镜像就类似于iso文件。比如说上操作系统课的时候，如果按照实验指导书的步骤安装centos操作系统，第一步你要做的就是上网下载下来centos的iso镜像文件，你下载成功之后，使用虚拟机软件基于你这个iso文件，创建一个运行在你虚拟机上

的操作系统。在这个例子之中，网上下载下来的那个iso文件就是镜像，而你根据iso文件创建出来的操作系统实例就是容器。

- 或者我们可以从用面向对象的角度看，Docker镜像就是类，它描述了容器的共同属性和行为。Docker容器可以看作是由Docker镜像实例化出来的对象。

在理解了docker容器与镜像，使用docker的各种命令就不会觉得很陌生了。在右边我也列出了许多常用的命令，可以发现这些命令大多都是针对于镜像和容器的，有了这些可以应对大部分场景了。不过docker命令有许多的参数可以选，这就需要RTFM了。

接下来我在自己的电脑上实际演示一下docker的基本操作，[（退出PPT）](#) 为了大家看的方便，我把常用命令贴图贴下来。

实例演示

官方文档展示

tutorial的部分

动手创建一个mysql容器

SSH连接创建好的容器