



链滴

架构师之路 -- 虚拟化技术与容器 Docker

作者: [jianzh5](#)

原文链接: <https://ld246.com/article/1592439357484>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

通常架构师们为了提高系统及硬件资源的利用率引入了虚拟化技术。

虚拟化是一种资源管理技术，它可以把各种实体资源抽象后再分隔，从而打破实体结构的限制，大程度的提高资源的利用率。

今天我们来聊一下传统虚拟化技术和以 Docker 为代表的容器技术。

虚拟化

虚拟化就是在台物理服务器上，运行多台“虚拟服务器”。这种虚拟服务器，也叫虚拟机 (V, Virtual Machine)。

虚拟化出现的初衷主要是为了减少物理服务器的数量，提高服务器资源利用率。利用虚拟化技术物理服务器隔离成多个小的虚拟服务器并提供部署能力。

简单来说，虚拟化可以为我们的工作带来以下几种便利：

-

- 减少物理服务器数量

- 提高资源利用率

- 可实现环境隔离

- 可实现资源隔离

- 虚拟机可实现分钟级交付

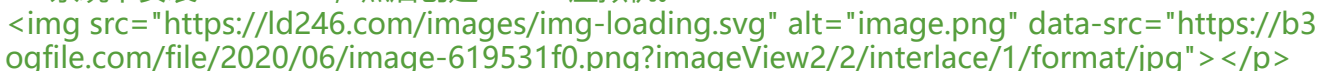
- 虚拟机的配件可动态扩容

- 虚拟机的计算节点可动态迁移

虚拟化技术

我们主要使用 Hypervisor，也叫做 VMM (Virtual Machine Monitor, 虚拟机监视器) 来实现服务器的虚拟化，Hypervisor 它不是一款具体的软件，而是一类软件的统称。像 VMware、KVM、Xe、Virtual Box，都属于 Hypervisor。

VMware 大家应该很熟悉，就是 VMware Workstation。学习 Linux 的话，很多人都是在 windows 系统下安装 VMware，然后创建 Linux 虚拟机。



Linux 服务器一般是使用 KVM 作为虚拟化工具，KVM (全称是 Kernel-based Virtual Machine) 是 Linux 下 x86 硬件平台上的全功能虚拟化解方案，包含一个可加载的内核模块 kvm.ko 提供和拟化核心架构和处理器规范模块。



虚拟化平台架构

虚拟化平台主要有以下三种架构：

-

- 本地存储




物理机上运行虚拟机，虚拟机的盘在物理机的磁盘上

- 集中存储



运行虚拟机的物理机，虚拟机的磁盘在共享存储中。在集中存储架构中如果一台 host 挂了，VM 可通过平台的设置去另外一台物理机上运行，这就实现了虚拟机的动态漂移。

- 分布式存储



分布式存储架构中虚拟机的磁盘位于所有服务器上，这种架构也叫“计算存储一体化”，虚拟机的磁会打散成很多块块，这些块块分布于集群内所有的机器上，这样就实现了存储能力的最大化。虚拟机的 IO 不再受限于单机的磁盘，而是利用到整个集群里磁盘的能力。削峰填谷、提高了虚拟机的 O 能力，而且通过冗余保障了数据的安全性。

这种架构的缺点是严重依赖网络稳定性，一旦网络挂了，所有的虚拟机就会全挂。

虚拟化平台的高可用机制

<p>虚拟化平台的高可用机制是指虚拟机所在的物理机挂了后可以将虚拟机快速切换到另外一台物理运行。

高可用的前提条件是物理机挂了，对于虚拟机内部的故障并没有检测机制，属于不完整的高可用方案主要是基于以下两个目标：</p>

虚拟机计算节点动态迁移

虚拟机磁盘动态迁移

<h3 id="虚拟化原则">虚拟化原则</h3>

<p>虚拟机的出现是为了使一些大的计算资源分割成许多小的资源然后灵活调配，遵循的是分而治的原则，如果一个虚拟机申请了物理机超过一半的资源就违背了这个原则，那样还不如直接使用物理来的干脆，所以我们在使用虚拟化的时候一般要遵循以下几个原则：</p>

某一虚拟机所占资源不超过宿主机的 40%

不承载磁盘 IO 密集型组件（数据库，消息队列，搜索引擎）

以双路 2U 服务器来说，整合比控制在 1: 4 – 1: 10 较为理想

CPU 可一定程度上超量分配，最多可以超分一倍左右

内存一般不可超量分配

<h2 id="Docker容器">Docker 容器</h2>

<p>在使用虚拟化一段时间后，发现它存在一些问题：</p>

虚拟机的系统层会占用比较多物理机的资源，需要更进一步提高服务器的资源利用率

当需要迁移虚拟机服务程序时，需要迁移整个虚拟机，迁移流程复杂

<p>为了解决这些问题，我们就引入了 容器。而大家常听说的 Dock r，就是创建容器的工具，是应用容器引擎。

容器也是虚拟化，但是属于“轻量级”的虚拟化。它的目的和虚拟机一样，都是为了创造“隔离环境”。但是，它又和虚拟机有很大的不同——虚拟机是操作系统级别的资源隔离，而容器本质上是进程级资源隔离。</p>

<p>

虚拟化 vs 容器</p>

<h3 id="虚拟化-VS-容器">虚拟化 VS 容器</h3>

<p>相比于传统的虚拟机，Docker 的优势很明显，它启动时间很快，是秒级，而且对资源的利用率高（一台主机可以同时运行几千个 Docker 容器）。此外，它占的空间很小，虚拟机一般要几 GB 到十 GB，而容器只需要 MB 级甚至 KB 级。</p>

<p>我们可以通过下面这张图来看虚拟化与容器之间的性能、资源利用率差距，非常明显</p>

<p><p>

<h3 id="容器编排">容器编排</h3>

<p>利用 Docker 可以很方便的创建容器，但是当容器数量达到一定规模，就需要编排工具去管理，是容器生命周期管理工具。</p>

<p>容器编排工具提供调度和管理集群的技术，提供用于基于容器应用可扩展性的基本机制。这些工使用容器服务，并编排他们以决定容器之间如何进行交互。</p>

<p>容器编排工具很多，有 Docker Swarm, Kubernetes, Mesos, Rancher, 下面用一张图来对下这些容器编排工具的特点及各自的优势。</p>

<p></p>

<h2 id="小结">小结</h2>

<p>今天主要对虚拟化技术及容器做了个简单的梳理，虚拟化和容器初衷与目的都是为了更好的提高

源利用率，至于两者之间的不同大家要务必记住一点：虚拟机是操作系统级别的资源隔离，而容器本上是进程级的资源隔离。 </p>