

# Docker

## Docker

1. 基本概念、用途、核心思想
2. Docker的组成、架构、重要概念
3. Docker的安装

# Docker

4. 第一个Docker容器

5. 运行Nginx镜像，并访问到Docker容器内部

6. 制作自己的Docker容器，`dockerfile`实战

## Docker的基本概念、用途、核心思想

◆ Docker应用广泛

◆ Docker是什么？

◆ 为什么需要Docker

# docker的基本概念、用途、核心思想

- ◆ Docker的用途
- ◆ Docker的特点
- ◆ Docker带来的好处

## docker应用广泛

### 阿里集团八年容器化演进之路

转载: Docker\_ 最后发布于2018-09-11 07:46:44 阅读量 491 ☆ 收藏



近日，阿里集团内部已经实现 100% 容器化镜像化；距离 PouchContainer 开源不到一年时间，PouchContainer 开源版 1.0 GA 版本发布，已经完全达到生产级别。另外，作为百万级开源容器技术，PouchContainer 被收录进为高校教材《云计算导论》。PouchContainer 现在服务于阿里巴巴集团和蚂蚁金服集团的绝大部分 BU，包括交易&中间件、B2B/ICBU/ICBU，搜索广告数据库，还有收购或入股的一些公司，比如优酷高德、UC等。其中体量最大的是交易和电商平台，在 2017 年双 11 的时候我们支撑了破纪录的峰值，背后的应用都是跑在 PouchContainer 里面，整体容器实例已经到了百万级规模。使用了 PouchContainer 的应用涵盖了各种各样的场景。这些场景从运行模式来看，有标准的在线 App，还有像购物车、广告、测试环境等比较特殊的场景。不同的场景对 PouchContainer 有不同的使用方式和需求。从编程语言看，实际运行着 Java、C/C++、Nodejs、GoLang 等语言编写的应用。从技术栈的角度看，包含了电商、DB、流计算、大数据、专有云等场景，每个场景对于容器各方面要求，所用到的特性都不太一样，PouchContainer 针对每个场景的需求都在产品上都做了支持。

Docker实践，来自沪江、滴滴、蘑菇街架构师的经验之谈

2019年07月09日 23:50 来源：未知 手机版



摘要：第一期主题：容器实践。Docker 作为当前最具颠覆性的开源技术之一，其轻量虚拟化、可移植性是CI/CD，DevOps，微服务的重要实现技术。但目前技术还不够成熟，在生产实践中会遇到不少坑。本期参与小组交流的是国内较早采用 Docker 实践的公司。

架构师小组交流会：每期选一个时下最热门的技术话题进行小组交流。

第一期主题：容器实践。Docker 作为当前最具颠覆性的开源技术之一，其轻量虚拟化、可移植性是CI/CD，DevOps，微服务的重要实现技术。但目前技术还不够成熟，在生产实践中会遇到不少坑。本期参与小组交流的是国内较早采用 Docker 实践的公司。

参与嘉宾：沪江架构师黄凯、滴滴架构师田智伟、蘑菇街架构师张振华、蘑菇街运维架构师向靖、七牛技术总监袁晓沛、扇贝技术总监丁彦

# Docker是什么？



- ◆ Docker是一个用来装程序及其环境的**容器**，属于Linux容器的一种封装，提供简单易用的容器使用接口。它是目前**最流行的**Linux容器解决方案。
- ◆ **比喻**：客车可以装人，衣柜可以放衣服

## 为什么需要Docker

- ◆ **环境配置**的难题
- ◆ 虚拟机
- ◆ Docker的基础——**Linux容器**

## Docker的用途

- ◆ 提供**统一**的环境
- ◆ 提供**快速拓展**、弹性伸缩的云服务
- ◆ 防止其他用户的进程把服务器资源**占用过多**

## Docker的特点

- ◆ **标准化**
  - ◆ 运输方式（把程序和环境从一个机器运到另一个机器的）
  - ◆ 存储方式（程序和环境的存储）
  - ◆ API接口（不需要Tomcat等应用的命令了，都标准化了）



## Docker的特点

- ◆ 灵活：即使是最复杂的应用也可以集装箱化
- ◆ 轻量级：容器利用并共享主机内核
- ◆ 便携式：可以在本地构建，部署到云，并在任何地方运行
- ◆ 可扩展：可以增加并自动分发容器副本

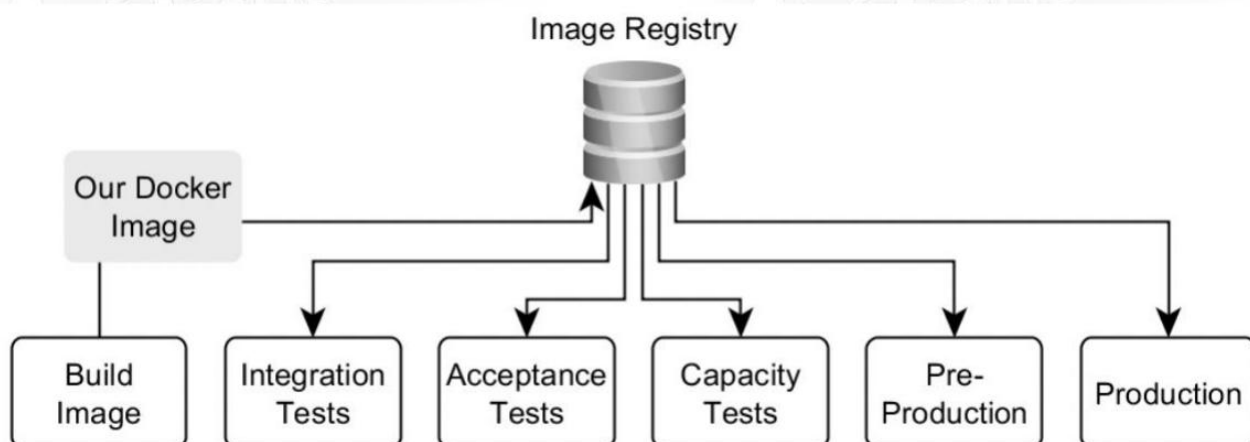
## Docker带来的好处

- ◆ 开发团队得到的好处

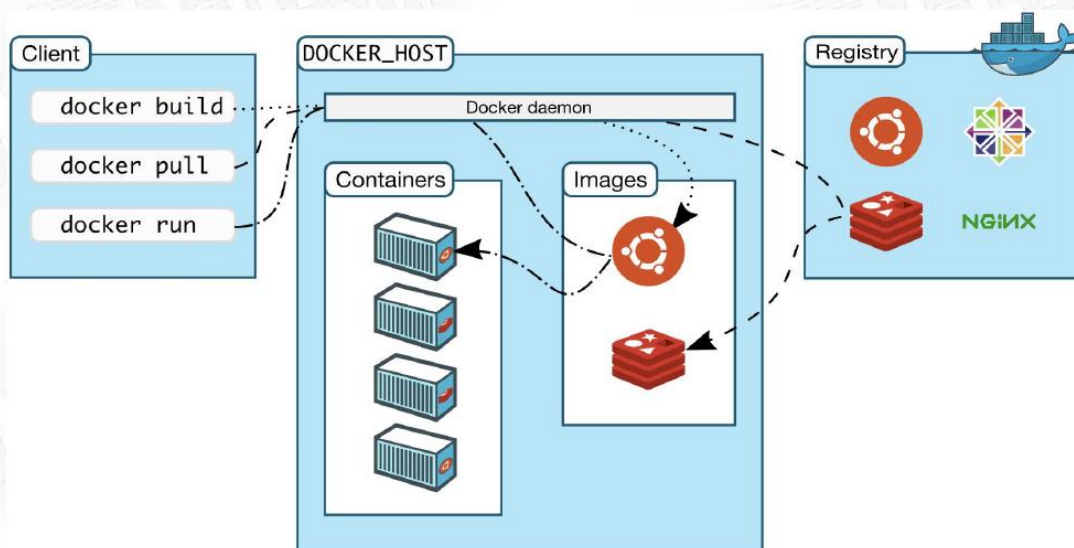
Our Docker Image	
JRE	<b>Our Application</b>
VM Arguments	Web Server/ Application Server (configured)
Environment Variables	
Files, Folders	Users, Permissions

# Docker带来的好处

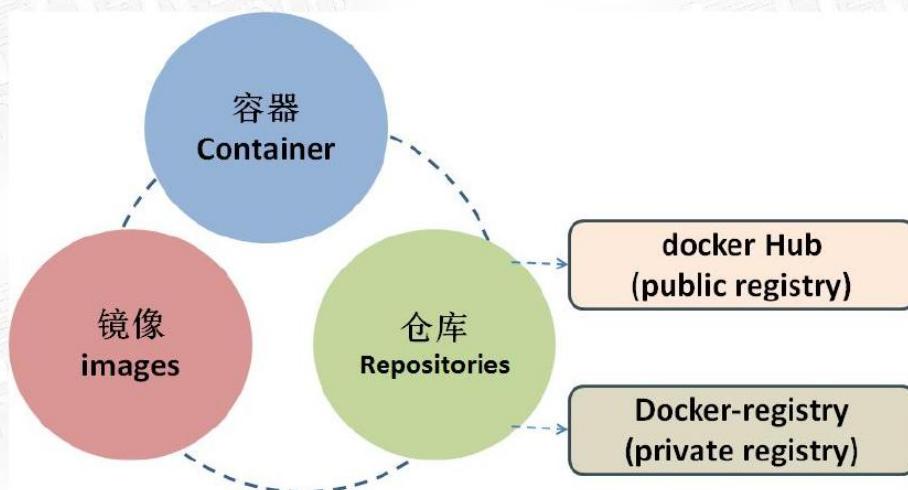
## ◆ 开发团队得到的好处



## Docker的组成、架构、重要概念



# Docker的组成、架构、重要概念

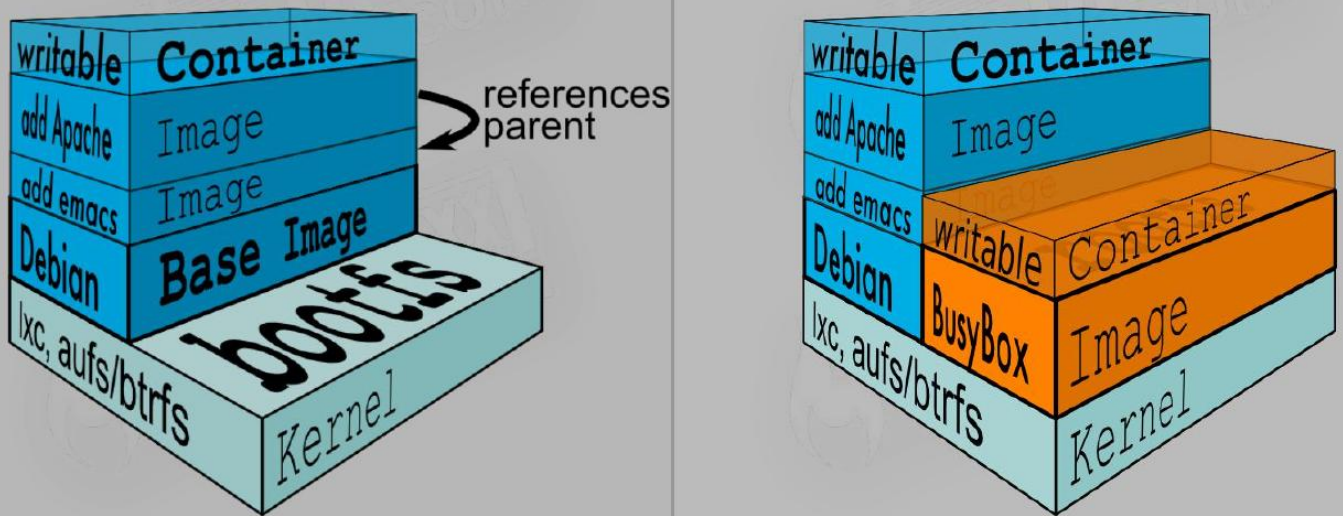


## image 镜像

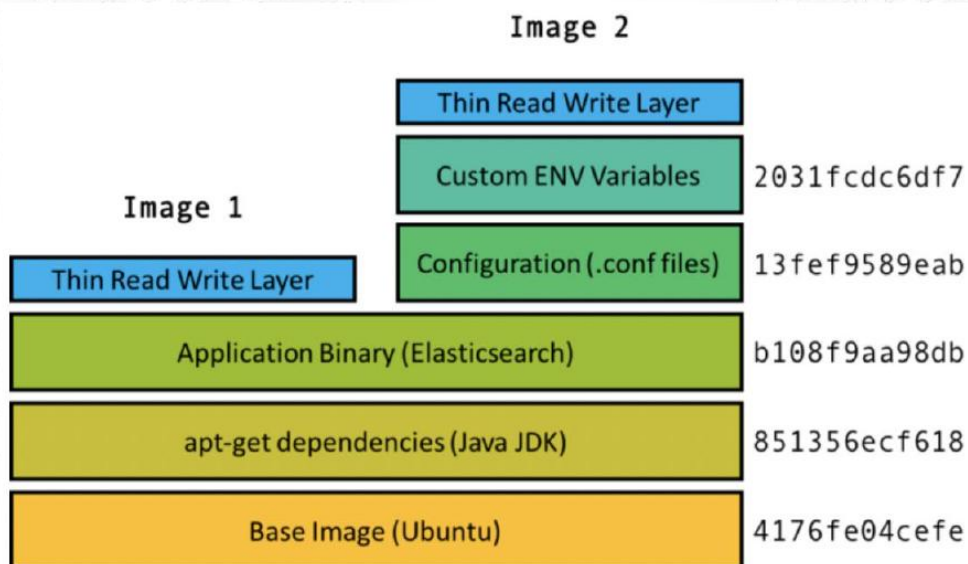
- ◆ 存储：联合文件系统，UnionFS
- ◆ 演示



## image 镜像



## image 镜像



# 为什么需要分布式

## ◆ 实际工作中的痛点

◆ 工程**臃肿**

◆ 测试、上线**繁琐**

◆ 开发**效率低**

## 容器

◆ 镜像类似于Java中的类，而容器就是实例

◆ 容器的这一层是可以修改的，而镜像是不可以修改的

◆ 同一个镜像可以生成多个容器独立运行，而他们之间没有任何的干扰

## 仓库

- ◆ `hub.docker.com`
- ◆ `https://c.163yun.com/hub#/m/home/`
- ◆ 公有、私有

## client和daemon

- ◆ **client** 提供给用户一个终端，用户输入 Docker 提供的命令来管理本地或远程的服务器
- ◆ **daemon**：服务端守护进程，接收 Client 发送的命令并执行相应的操作

# Docker的安装

- ◆ Mac OS
- ◆ Linux (Cent OS)
- ◆ 实操演示

## 第一个Docker容器

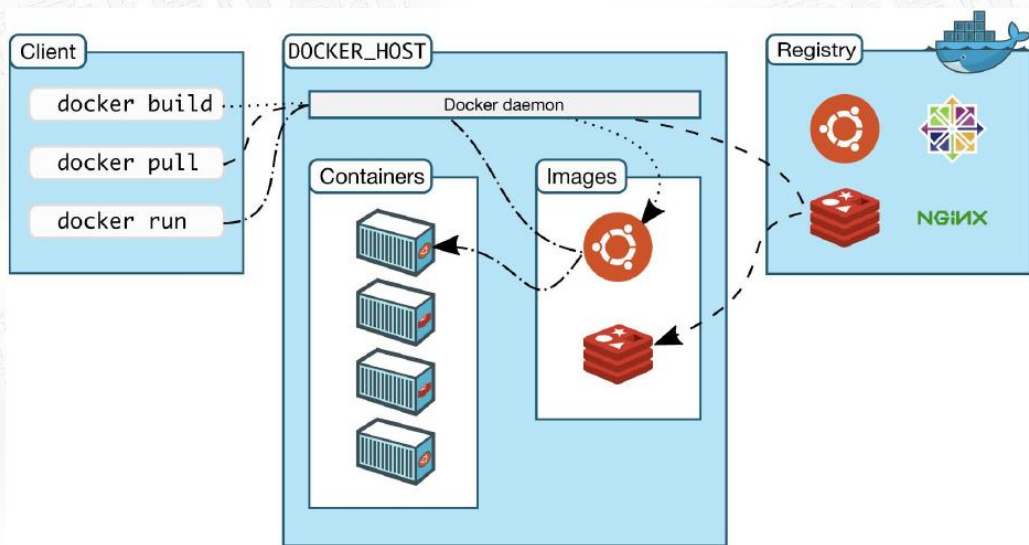
- ◆ 下载镜像
- ◆ `docker pull [OPTIONS] NAME[:TAG]`
- ◆ `docker images [OPTIONS] [REPOSITORY[:TAG]]`

# 第一个Docker容器

## ◆ 运行镜像

◆ `docker run [OPTIONS] IMAGE [COMMAND] [ARG...]`

# 第一个Docker容器



# 运行Nginx镜像，并访问到Docker容器内部

- ◆ 前台、后台
- ◆ 拉取并运行
- ◆ 三种网络模式

## Docker的网络模式

- ◆ Bridge
- ◆ Host
- ◆ None
- ◆ 端口映射



## 实操

- ◆ 访问Docker内的Nginx

## 制作自己的Docker容器，dockerfile实战

- ◆ dockerfile作用
- ◆ 使用dockerfile的好处
- ◆ 写自己的dockerfile

# 制作自己的Docker容器，dockerfile实战

FROM alpine:latest

MAINTAINER imooc

CMD echo 'hello my dockerfile'

## build生成自己的镜像

- ◆ docker build -t hello\_docker .
- ◆ 运行自己的镜像

## 总结： Docker

1. 基本概念、用途、核心思想
2. Docker的组成、架构、重要概念
3. Docker的安装

## 总结： Docker

4. 第一个Docker容器
5. 运行Nginx镜像，并访问到Docker容器内部
6. 制作自己的Docker容器， dockerfile实战